

THESIS / THÈSE

MASTER EN SCIENCES MATHÉMATIQUES

Modélisation des déménagements dans la plateforme VirtualBelgium

RAMELOT, Elodie

Award date:
2014

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



UNIVERSITE DE NAMUR

Faculté des Sciences

**Modélisation des déménagements dans
la plateforme VirtualBelgium**

**Mémoire présenté pour l'obtention
du grade académique de master en Sciences Mathématiques à finalité spécialisée**

Elodie RAMELOT

Août 2014



UNIVERSITE DE NAMUR

Faculté des Sciences

**Modélisation des déménagements dans
la plateforme VirtualBelgium**

**Mémoire présenté pour l'obtention
du grade académique de master en Sciences Mathématiques à finalité spécialisée**

Elodie RAMELOT

Promoteur : Philippe TOINT

Co-promoteur : Eric CORNELIS

Août 2014

Tout d'abord, j'aimerais remercier l'ensemble des membres du département de Mathématiques de l'Université de Namur pour nous avoir enseigné de nouveaux concepts et théorèmes, mais aussi pour l'expérience incroyable que j'ai vécue durant ces 5 dernières années.

Je désire remercier plus particulièrement mon promoteur Philippe Toint, mon co-promoteur Eric Cornelis et les deux assistants Johan Barthelemy et Laurie Hollaert sans qui ce mémoire n'aurait pas été possible.

Enfin, j'aimerais exprimer ma reconnaissance envers mon entourage et les membres de ma famille qui m'ont dévoué un soutien absolu lors de ce travail.

Résumé :

Le projet VirtualBelgium est une plateforme de simulation de l'évolution d'une population, créée par le Groupe de Recherche sur les Transport (GRT-naXys). Actuellement, cette plateforme est appliquée à la Belgique, où une population synthétique d'environ 10 millions d'individus répartis en plus de 4 236 000 ménages évolue temporellement mais également spatialement dans un réseau virtuel proche de celui de notre pays. Cette simulation est un outil qui permet de mieux comprendre l'évolution de la population belge à travers différentes problématiques dont la démographie et la mobilité et ainsi mieux gérer leurs conséquences. Le développement de cette plateforme passe par la modélisation du choix de migration résidentielle d'un individu. En effet, les déménagements des individus ne font pas encore partie du processus d'évolution de la population. Ce mémoire consiste donc à implémenter un modèle issu du projet MOBLOC, qui fournira la propension à migrer d'un individu ainsi que la nouvelle commune de résidence de celui-ci, en tenant compte des caractéristiques des individus et des ménages définies dans VirtualBelgium.

Abstract :

The VirtualBelgium project is a simulation platform of the evolution of a population, created by « le Groupe de Recherche sur les Transport (GRT-naXys) » of the University of Namur. For this moment, this platform is applied on Belgium. A synthetic population about 10 million individuals distributed in around 4 236 000 households evolves over time but also spatially in a virtual network close to the actual Belgian one. This simulation is a tool which allows to better understand the evolution of the Belgian population and considers various problems as the demography or the mobility and so better to manage their consequences. The development of this platform requires the modelling of the choice of residential migration for an individual. Indeed, migrations of individuals are not already included in the evolution process of the population. This thesis consists in the creation of a model resulting from MOBLOC project, which will give the propensity to move for an individual and also his residential municipality, taking into account individuals and households attributes defined by VirtualBelgium.

Table des matières

Introduction	1
1 Présentation de VirtualBelgium	3
1.1 Génération d'une population synthétique	3
1.2 Les logiciels utilisés	4
1.2.1 Repast HPC	5
1.2.2 MATSim	5
1.3 Coeur du programme	6
1.3.1 Classes générales	6
1.3.2 Individus et Ménages	7
1.3.3 Réseau	8
1.4 Programmation parallèle	9
2 Évolution temporelle	11
2.1 Caractéristiques de la population synthétique	11
2.1.1 Individus	12
2.1.2 Ménages	15
2.2 Processus de l'évolution temporelle	16
2.2.1 Évolution de l'âge	16
2.2.2 Naissance	16
2.2.3 Décès	17
2.2.4 Autres processus	17
2.3 Migration résidentielle	18
3 MOBLOC : présentation des modèles	19
3.1 Objectif	19
3.2 Méthodologie	19
3.3 Modèles de transport	21
3.3.1 Modèle de demande en mobilité	21
3.3.1.1 Données	22
3.3.1.2 Méthodologie	22
3.3.1.3 Modèle gravitaire	22
3.3.2 Modèle de répartition modale (Modal split)	24

3.3.3	Modèle d'accessibilité	26
3.3.3.1	Modèle pour la voiture	26
3.3.3.2	Modèle pour les transports en commun	29
3.4	Modèle de migration résidentielle	29
3.4.1	Modèle de propension à migrer	30
3.4.1.1	Sélection des variables	30
3.4.1.2	Méthodologie	32
3.4.1.3	Modèle	32
3.4.2	Modèle de localisation	34
3.4.2.1	Méthodologie	34
3.4.2.2	Modèle	35
3.5	Modèles d'évolution	36
4	Migration résidentielle	37
4.1	Modèle de propension à migrer	37
4.1.1	Régression logistique binaire	37
4.1.2	Variables explicatives	40
4.1.2.1	Âge de l'individu	40
4.1.2.2	Genre de l'individu	42
4.1.2.3	Nationalité de l'individu	43
4.1.2.4	Type et taille du ménage	43
4.1.2.5	Évolution du type et de la taille du ménage	46
4.1.2.6	Position de l'individu au sein du ménage	49
4.1.2.7	Évolution de la position de l'individu	51
4.1.2.8	Migration précédente	53
4.1.2.9	Niveau d'éducation de l'individu	55
4.1.2.10	Type d'occupation du logement	56
4.1.3	Implémentation de la régression	57
4.2	Modèle de localisation	59
4.3	Résultats	62
4.3.1	Migrations	62
4.3.2	Migrations en fonction de la classe d'âges	63
4.3.3	Migrations en fonction du genre	65
4.3.4	Migrations en fonction du niveau d'éducation	66
4.3.5	Migrations en fonction de l'obtention du permis de conduire	67
4.3.6	Migrations en fonction de la position dans le ménage	68
	Conclusion	69
	Bibliographie	71

A	Installation de VirtualBelgium	75
A.1	Installation	75
A.2	Téléchargements	75
A.3	Repast HPC	77
A.4	Compilation et exécution de VirtualBelgium	78
A.5	Configuration de VirtualBelgium	79
A.6	Mise à jour du code	79
B	Concepts de la programmation orientée objet	81
C	Implémentation du modèle de migration résidentielle	83
C.1	Modèle de propension à migrer	83
C.1.1	Implémentation de l'âge	83
C.1.2	Implémentation du genre	84
C.1.3	Implémentation de la nationalité	84
C.1.4	Implémentation du type et de la taille du ménage	84
C.1.5	Implémentation de l'évolution du type et de la taille du ménage	86
C.1.6	Implémentation de la position au sein du ménage	93
C.1.7	Implémentation de l'évolution de la position au sein du ménage	94
C.1.8	Implémentation de la migration précédente	96
C.1.9	Implémentation du niveau d'éducation	96
C.1.10	Implémentation du type d'occupation du logement	97

Introduction

De nos jours, l'évolution de la population est un sujet qui mérite d'être traité avec importance. Dans le but de mieux comprendre son comportement, nous avons besoin de construire des outils de simulation.

Mon mémoire se situe dans le cadre du projet VirtualBelgium dans lequel travaillent Laurie Hollaert et Johan Barthélemy, qui a été créé par le Groupe de Recherche sur les Transports (GRT-naXys) en collaboration avec des membres de l'Université de Namur tels que Philippe Toint et Éric Cornélis. Celui-ci consiste à établir l'évolution d'une population synthétique composée d'individus et de ménages et construite pour la Belgique. Pour faire évoluer cette population dans le temps, il faut prendre en compte l'évolution de caractéristiques telles que l'âge, le diplôme, le type du ménage, . . . et simuler les événements qui peuvent se produire tels que les naissances et décès, les mariages et divorces, les déménagements, etc.

Dans le cadre de ce mémoire, je serai amenée à simuler les déménagements des ménages. Autrement dit, je devrai prédire le fait de déménager ou non ainsi que le fait de déménager dans la commune de résidence actuelle ou hors de celle-ci. Pour ce faire, je devrai me réapproprier les idées déjà établies dans les deux rapports publiés [2, 7] de MOBLOC, projet de recherche qui avait pour but d'analyser les choix de localisation des ménages, leur accessibilité journalière ainsi que les interactions entre ces deux éléments.

Dans le premier chapitre, nous présenterons les différents éléments qui composent la plateforme VirtualBelgium ainsi que la manière dont la population synthétique a été générée. Les logiciels utilisés par la plateforme ainsi que leurs caractéristiques seront brièvement décrits. Ensuite, nous aborderons l'architecture de ce programme et nous énoncerons les différentes caractéristiques qui définissent les individus et les ménages de la population synthétique.

Le but de ce mémoire étant de simuler la migration résidentielle, nous aurons besoin de connaître les différentes caractéristiques des individus et des ménages définies dans VirtualBelgium. En effet, celles-ci joueront un rôle essentiel dans le choix de migration. L'évolution de la population synthétique se situe à deux niveaux différents : l'évolution spatiale modélisée par les chaînes d'activités des individus au cours d'une journée, et l'évolution temporelle modélisée par des événements se déroulant sur une année. Dans notre cas, nous allons plutôt nous attarder sur cette deuxième évolution et sur ses différents processus implémentés.

Le choix de résidence fait par les ménages, la demande de transport et l'évolution de l'accessibilité qui en résulte ont été étudiés dans le projet MOBLOC. Nous allons donc décrire chacun de ces modèles afin de mieux comprendre de quoi il s'agit et de connaître les variables qui vont être nécessaires à l'implémentation de la migration résidentielle dans VirtualBelgium.

Enfin, dans le dernier chapitre, nous expliquerons les différentes étapes de l'implémentation du modèle de migration résidentielle. Nous tenterons d'établir des correspondances entre les variables issues de MOBLOC et celles définies dans VirtualBelgium. Les résultats obtenus après avoir appliqué ce modèle à la population synthétique seront exposés pour clôturer ce mémoire.

Pour terminer, une conclusion reprendra les points essentiels abordés dans ce travail, les modifications apportées ainsi que les améliorations futures qui mériteraient d'être apportées au programme.

Chapitre 1

Présentation de VirtualBelgium

La plateforme de simulation VirtualBelgium a pour ambition de modéliser l'évolution d'une population via une simulation de différents aspects : démographie, choix résidentiel, mobilité, . . . Ce projet est mené par le Groupe de Recherche sur les Transports (GRT-naXys) et constitue une partie de la thèse de J. Barthélemy [4]. Cet outil est actuellement appliqué à la Belgique.

Dans cette section, nous expliquons les différents éléments qui composent la plateforme mais également les différentes données nécessaires à son utilisation. De plus, à partir de maintenant, nous considérerons uniquement l'outil appliqué à la Belgique.

1.1 Génération d'une population synthétique

Avant de pouvoir simuler une quelconque évolution avec l'outil VirtualBelgium, ce dernier a besoin d'une population initiale. Malheureusement, pour des raisons de confidentialité et budgétaires, il n'est pas possible de récupérer les données personnelles attendues concernant chaque citoyen belge. Il faut donc générer une population synthétique qui doit être aussi statistiquement proche que possible de la population réelle belge.

Cette population synthétique a été générée par J. Barthélemy et Ph. Toint selon une nouvelle méthode qu'ils proposent dans l'article [3]. Celle-ci est composée de 10.262.160 individus répartis en 4.236.202 ménages dans les 589 communes belges. Les données utilisées au cours du processus concernent la population belge en 2001 et proviennent des sources suivantes :

- *Directorate-general Statistics and Economic information* du gouvernement fédéral belge (2001)
- *Service public fédéral Mobilité et Transports* du gouvernement fédéral belge (2000)
- *GéDAP*¹ centre de recherches de l'université catholique de Louvain (2001)
- L'enquête de mobilité *MOBEL* (1999) [16]

1. Groupe d'étude de démographie appliquée

Ensuite, l'évolution de la population synthétique est prise en charge par un programme implémenté en C++ dont la structure est présentée à la figure 1.1. Ce programme permet de simuler une évolution temporelle (démographie) et spatiale (mobilité) de la population synthétique, variant selon les données qui lui sont fournies et utilisant le framework Repast HPC, expliqué dans la section suivante.

Après l'exécution du programme, des outputs tels que des statistiques concernant l'évolution ou encore des fichiers contenant les déplacements des individus sont créés. Ces derniers fichiers sont ensuite exploités par le logiciel MATSim qui permet une visualisation du trafic routier. Ce logiciel est également présenté dans la section suivante.

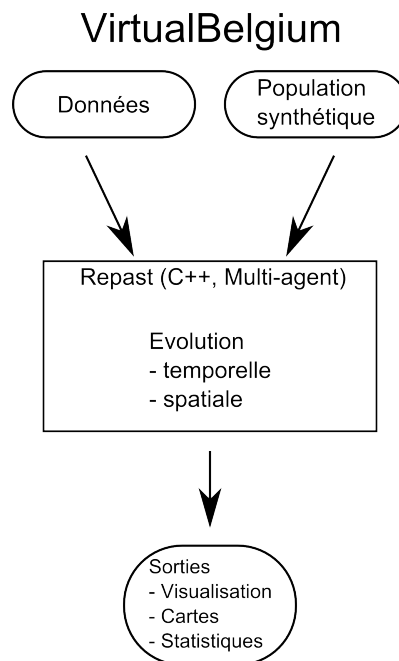


FIGURE 1.1 – Structure de VirtualBelgium

La suite de ce chapitre permet de comprendre le fonctionnement du programme VirtualBelgium lié à la plateforme. Il est donc conseillé de connaître certaines notions de langages de programmation orientée objet. Pour mieux appréhender ces notions, quelques principes de base de la programmation orientée objet sont rappelés dans l'annexe B.

1.2 Les logiciels utilisés

Comme nous venons de le dire dans la section précédente, les deux logiciels principalement utilisés par le programme VirtualBelgium sont Repast HPC et MATSim. Leur utilisation et leurs principales caractéristiques sont décrites ci-dessous.

1.2.1 Repast HPC

Repast HPC (REcursive Porous Agent Simulation Toolkit for High Performance Computing) est un framework C++ utilisé pour la simulation et la modélisation de systèmes multi-agents. D'après la référence [14], on appelle multi-agents, un système où plusieurs agents évoluent et qui possède des caractéristiques telles que le parallélisme², la robustesse³ et l'extensibilité⁴. Cependant, le terme « agent » peut être interprété de plusieurs manières en informatique. Dans notre cas, un agent est un système informatique situé dans un environnement qu'il peut modifier, qui agit de manière autonome et flexible pour atteindre les objectifs qui lui sont fixés. Les agents sont capables d'interagir entre eux.

Repast est un outil gratuit, open source et une plate-forme de modélisation et simulation multi-agents. Cet outil est surtout utilisé pour des applications de sciences sociales. À l'origine, il a été développé par David Sallach ainsi que d'autres chercheurs de l'Université de Chicago et du Laboratoire National D'Argonne. Désormais, il est dirigé par l'organisation de bénévoles ROAD (Repast Organization for Architecture and Development), d'après la référence [24].

Son architecture permet de représenter chaque agent par un identifiant unique et prend en charge la gestion des opérations parallèles grâce aux bibliothèques MPI et Boost. Dans VirtualBelgium, les individus et les ménages de la population synthétique sont représentés par des agents.

Vous trouverez en annexe à la page 75, un manuel d'utilisation de VirtualBelgium. Il comprend également tout le processus d'installation de Repast HPC sur un ordinateur personnel ou sur un « cluster » pour des calculs intensifs, ainsi que toutes les informations relatives à la compilation et au débogage.

1.2.2 MATSim

MATSim (Multi-Agent Transport Simulation) est un outil de simulation de transport. Il s'agit d'un projet open source principalement créé par les groupes suivants [20] :

- *Transport Systems Planning and Transport Telematics*, Institute for Land and Sea Transport Systems, Technische Universität Berlin
- *Transport Planning*, Institute for Transport Planning and Systems, Swiss Federal Institute of Technology Zurich
- *Senozon*, entreprise suisse avec une filiale en Allemagne

2. « En informatique, le parallélisme consiste à implémenter des architectures d'électronique numérique permettant de traiter des informations de manière simultanée, ainsi que les algorithmes spécialisés pour celles-ci. Ces techniques ont pour but de réaliser le plus grand nombre d'opérations pour réduire le temps d'exécution. » [29]

3. « La robustesse d'un programme est entre autres, sa capacité à bien fonctionner. » [12]

4. « Un système est extensible si on peut étendre ses capacités par ajout de mises à jour ou de modules. » [12]

Dans VirtualBelgium, MATSim est utilisé pour créer le réseau utilisé par le programme, à partir des données extraites du site OpenStreetMap [22], puis pour réaliser l'affectation des flux de trafic sur ce réseau. *OpenStreetMap est une carte du monde librement modifiable, collaborative où les données sont libres d'être téléchargées et utilisées, sous les termes d'une licence ouverte.* Une section est consacrée au réseau à la fin de ce chapitre.

1.3 Coeur du programme

Intéressons-nous à présent à l'architecture du programme VirtualBelgium. Celle-ci est représentée en annexe, à la figure A.1, qui décrit les différentes classes implémentées. Une classe est un principe utilisé dans la programmation orientée objet. Elle est constituée d'un ensemble d'attributs et de fonctions, appelées méthodes.

Sur cette figure, chaque cadre correspond à une classe du programme dont le nom est inscrit en haut de ce cadre. Ce dernier est divisé en deux parties : la première contient les attributs de la classe et la deuxième les méthodes principales. De plus, des liens indiquent les relations éventuelles existantes entre les différentes classes.

Parcourons les différentes classes et résumons leurs rôles et leurs principales caractéristiques.

1.3.1 Classes générales

La classe **Data** est chargée de lire et récupérer toutes les données nécessaires au programme VirtualBelgium. Les données utilisées sont les suivantes :

- *propriétés des modèles* telles que l'évolution spatiale ou temporelle ;
- *données socio-démographiques* : pyramide des âges par commune pour les hommes et les femmes, probabilité de naissance par âge, probabilité de décès par âge et sexe, probabilité d'avoir un garçon ou une fille ;
- *données relatives au réseau* : noeuds représentant chaque commune et réseau routier ;
- *données relatives au modèle d'activités* : codes pour les activités, paramètres de distribution concernant les activités (distance, durée du voyage, durée de l'activité et heure de départ et d'arrivée).

La classe **RandomGenerators** contient des générateurs⁵ de nombres pseudo-aléatoires suivant une loi de probabilité, utilisés tout au long du programme.

La classe **Model** se charge principalement de l'initialisation du modèle et décrit la procédure d'évolution que subit la population via la méthode **step**. Elle génère également les individus et

5. Un générateur de nombres pseudo-aléatoires est un outil permettant de ressortir une suite de nombres suivant une loi demandée à partir d'un nombre qu'on appelle graine (seed).

les ménages, ainsi que leurs caractéristiques qui sont détaillées ci-dessous. La méthode **step** est utilisée pour itérer sur tous les agents du modèle afin qu'ils accomplissent les tâches demandées par le modèle, telles que vieillir, mourir, déménager, se rendre à leurs activités de la journée, se marier ou divorcer, ...

1.3.2 Individus et Ménages

VirtualBelgium comporte deux différents types d'agents : les ménages représentés par la classe **Household** et les individus par la classe **Individual**. À chaque agent sont associées plusieurs attributs ainsi qu'un identifiant unique, déterminé par le logiciel Repast.

Un ménage contient les identifiants de chaque individu le composant, l'identifiant du noeud du réseau correspondant au domicile du ménage ainsi que le code INS⁶ de sa commune. La classe **Household** contient également des attributs concernant le nombre d'enfants (de 0 à 5), le nombre d'adultes supplémentaires (de 0 à 2) et le type du ménage parmi :

- Homme seul sans enfant
- Femme seule sans enfant
- Homme seul avec enfant(s) (et adulte(s) supplémentaire(s))
- Femme seule avec enfant(s) (et adulte(s) supplémentaire(s))
- Couple sans enfant
- Couple avec enfant(s) (et adulte(s) supplémentaire(s))

Un individu est décrit par son genre (féminin ou masculin), son âge et la classe d'âge associée, sa catégorie professionnelle (actif, inactif ou étudiant), son niveau d'éducation (aucun diplôme, diplôme primaire, diplôme secondaire ou diplôme supérieur), l'identifiant du ménage dont il fait partie et la place qu'il y occupe (chef de ménage, conjoint, enfant ou adulte supplémentaire) ainsi que son obtention ou non du permis de conduire.

De plus, les individus possèdent également une chaîne d'activités, c'est-à-dire une liste d'activités qu'ils doivent effectuer pendant leur journée. Par exemple, l'agenda d'un étudiant peut se résumer à deux activités : la première étant l'école et la deuxième, les loisirs. Les différentes activités possibles sont reprises dans la table 1.1.

6. « Le code INS est un code à 5 chiffres attribué à chaque commune par l'Institut National de la Statistique. » [21]

NUMÉRO	ACTIVITÉ
1	Déposer / reprendre quelqu'un
2	Activités à la maison
3	Se déplacer pour le travail (aller voir des clients par exemple)
4	Travail
5	École
6	Manger à l'extérieur
7	Faire les courses
8	Activités personnelles (banque, docteur)
9	Rendre visite à la famille / aux amis
10	Promenade
11	Loisirs
12	Autre

TABLE 1.1 – Les activités dans VirtualBelgium

Toute activité d'un individu se voit associer un lieu et une durée. Une telle modélisation par chaînes d'activités permet de simuler les déplacements des individus dans le réseau, en d'autres mots leur évolution spatiale. Les chaînes d'activités nous permettent donc d'analyser la mobilité belge en vue de mieux la comprendre.

1.3.3 Réseau

Le réseau utilisé dans VirtualBelgium est le réseau routier belge entier constitué de tous les carrefours et toutes les routes du pays et extrait du site OpenStreetMap. Le réseau est ensuite modifié par OSMOSIS⁷ pour conserver uniquement les données nécessaires au programme VirtualBelgium correspondant au réseau routier belge avec comme conséquence, par exemple, que les noeuds représentant les gares dans OpenStreetMap ne sont pas pris en considération. Ensuite, ce réseau est converti dans un type de fichier XML exploitable par VirtualBelgium et MATSim.

Dans le programme, le réseau est représenté par un *graphe*. Un graphe est un objet mathématique constitué de deux ensembles finis : le premier, noté V , contient n points appelés *noeuds* ou *sommets* labellisés de 1 à n ; le deuxième, noté E , contient des liens entre certaines paires de noeuds. Un lien entre deux noeuds est appelé *arc* et est noté (i, j) lorsqu'il relie les noeuds i et j .

Notre réseau est constitué d'environ 262.000 noeuds et 830.000 arcs. Les noeuds et arcs représentent respectivement les carrefours du réseau et les routes. Les arcs ne sont pas orientés.

7. OSMOSIS est une application java pour la manipulation des données OSM, extraites d'OpenStreet-Map.

1.4 Programmation parallèle

Étant donné la quantité de données traitées dans VirtualBelgium avec ses 10.262.160 individus et ses 4.326.202 ménages, ce programme ne peut s'exécuter sur une unique machine possédant un seul processeur⁸. En effet, si on exécute le programme uniquement en se restreignant à la population de Namur sur un ordinateur personnel avec deux processeurs, il faut déjà attendre deux jours pour obtenir les résultats. Ce temps important de calcul est en grande partie dû aux chaînes d'activité des individus. Si nous choisissons de ne pas assigner d'activités aux individus, le programme va s'exécuter quasiment de manière instantanée si on choisit de regarder l'évolution de la population sur une année.

Une solution à ce problème est la programmation parallèle, traitable par le logiciel Repast HPC et par la norme MPI (Message Passing Interface). Ce type de programmation consiste à effectuer différentes tâches en même temps grâce à l'utilisation de plusieurs processeurs. L'utilisation de la norme MPI permet d'exploiter les ordinateurs multiprocesseurs en assignant l'exécution d'une suite séquentielle d'opérations à chaque processeur. Cette assignation est réalisée en utilisant des opérateurs d'envoi et de réception de messages. Chaque processeur possède son propre espace mémoire. Pour obtenir une interaction entre les processeurs, il existe des messages spécifiques relatifs à la synchronisation.

Dans le programme VirtualBelgium, chaque processeur va s'occuper de son propre ensemble de ménages et donc de son propre ensemble d'individus composant ces ménages. Actuellement, aucune synchronisation n'est effectuée entre ces processeurs, il n'y a donc pas de lien entre les ménages de chaque processeur. Le réseau de VirtualBelgium n'est pas commun. En effet, chaque processeur possède son propre réseau, ce qui veut dire que si on modifie un noeud dans un processeur, il n'est pas modifié dans le réseau des autres processeurs.

En conclusion, il faut garder en mémoire le fait que le programme VirtualBelgium est adapté à la programmation parallèle et que chaque processeur possède son propre ensemble d'individus et son propre réseau.

Cette introduction au projet VirtualBelgium a été écrite en collaboration avec W. Henrotin, S. Marchal et G. Picard.

8. Un processeur est la partie centrale d'un ordinateur qui effectue les opérations arithmétiques et logiques. Cette définition a été inspirée du site de FUTURA- SCIENCES [13].

Chapitre 2

Évolution temporelle

VirtualBelgium dispose d'outils permettant d'étudier et de comprendre l'évolution du comportement de la population belge. Cette évolution se situe à deux niveaux différents. D'une part, l'évolution spatiale des individus est modélisée par leurs chaînes d'activités qui se déroulent sur une journée. D'autre part, l'évolution temporelle des individus est modélisée par des événements qui se déroulent sur une année (d'une année à la suivante) tels que le vieillissement, le décès, la naissance, la migration, le mariage ou le divorce, . . . Si l'on combine la simulation de ces deux évolutions, on peut aussi modéliser les interactions entre mobilité journalière et mobilité résidentielle.

Dans ce chapitre, nous allons tout d'abord donner les caractéristiques qui définissent les individus et les ménages de cette population. Le but de ce travail étant de modéliser la migration résidentielle, celles-ci joueront un rôle essentiel dans le choix résidentiel des individus lors d'un déménagement. Ensuite, nous décrirons les différents processus qui font actuellement partie de l'évolution temporelle de la population synthétique générée par VirtualBelgium. Nous terminerons par le processus qui nous intéresse et qui ne fait pas encore partie de la plateforme, c'est-à-dire la migration résidentielle.

2.1 Caractéristiques de la population synthétique

Comme nous l'avons expliqué dans le chapitre précédent, la population synthétique générée par VirtualBelgium représente la population de la Belgique en 2001 et est composée de 10.262.160 individus répartis en 4.236.202 ménages dans les 589 communes belges.

Chaque individu constitue un agent avec ses propres caractéristiques et il en va de même pour chaque ménage. Celles-ci ont été brièvement expliquées dans le chapitre précédent à la section 1.3.2. Nous allons détailler plus précisément celles dont nous aurons besoin par la suite lors de l'implémentation du modèle de migration résidentielle.

2.1.1 Individus

Chaque individu de la population synthétique est caractérisé dans VirtualBelgium par une liste d'attributs. Ceux-ci sont initialisés à partir des données collectées ou à partir d'autres attributs. Nous allons les expliquer ainsi que les différentes valeurs qu'ils peuvent prendre.

Identifiant

Un individu est tout d'abord caractérisé par un identifiant unique ainsi qu'un identifiant correspondant au ménage auquel il appartient. Il s'agit d'un nombre entier qui sert à identifier un individu en particulier. Ce nombre peut prendre une valeur comprise entre 1 et 10.262.160 s'il représente un individu ou comprise entre 1 et 4.236.202 s'il représente un ménage.

Cependant, l'identifiant d'un individu peut être différent lorsque ce dernier ne fait pas partie de la population synthétique de base. En effet, lorsqu'un bébé naît, il se voit attribuer un identifiant qui commence à 21.000.000 et qui augmente de 1 pour chaque nouvelle naissance.

Genre

Une caractéristique fondamentale d'un individu est bien entendu son genre, masculin ou féminin. Ce dernier est représenté par une variable de type « caractère » qui vaut :

- 'H' : homme
- 'F' : femme

Lorsque se produit une naissance, on attribue un genre au nouveau-né en fonction de la probabilité pour la mère d'avoir un garçon ou une fille, ce qui dépend, entre autres, de l'âge de la mère.

Âge

Un individu est également caractérisé par une classe d'âges qui provient des données collectées pour la création de la population synthétique. Les 5 classes d'âges sont représentées par des nombres entiers de la manière suivante :

- 0 : de 0 à 5 ans
- 1 : de 6 à 17 ans
- 2 : de 18 à 39 ans
- 3 : de 40 à 59 ans
- 4 : 60 ans et plus

Pour chaque individu, l'âge est calculé sur base de la classe d'âges à laquelle il correspond, ainsi qu'à partir de la pyramide des âges de la commune où il habite. Naturellement, l'âge d'un nouveau-né est initialisé à 0 l'année de sa naissance et la classe d'âges correspondante est 0.

Lieu de domicile

Le lieu de domicile d'un individu est représenté par le code INS de sa commune ainsi que par l'identifiant du noeud du réseau correspondant. Le code INS est fourni grâce aux données

collectées pour la population, tandis que le noeud représentant le domicile de l'individu est choisi aléatoirement parmi les noeuds du réseau situés dans sa commune de résidence.

Lors d'une naissance, le nouveau-né se voit attribuer le même identifiant de noeud du réseau que sa mère pour représenter son domicile. Il en va de même concernant le code INS de sa commune de résidence.

Niveau d'éducation

Le niveau d'éducation de l'individu représente le plus haut diplôme que celui-ci a obtenu. Il existe 4 niveaux différents dans VirtualBelgium qui sont représentés par des variables de type « caractère ». Ces dernières prennent les valeurs suivantes :

- 'O' : aucun diplôme
- 'P' : diplôme d'école primaire
- 'S' : diplôme d'école secondaire
- 'U' : diplôme d'école supérieure

Étant donné qu'un individu lors de sa naissance n'a pas encore pu aller à l'école et n'a donc aucun diplôme, son niveau d'éducation est initialisé à 'O'.

Statut professionnel

Une autre caractéristique d'un individu concerne le statut professionnel de celui-ci. Les 4 statuts existants sont également représentés par des variables de type « caractère » qui ont pour valeur :

- 'I' : inactif
- 'E' : étudiant
- 'A' : actif
- 'X' : aucun statut

Le statut professionnel n'est pas initialisé lors de la naissance d'un individu. Ce dernier ne possède donc aucun statut, ce qui est représenté par 'X'. Cependant, j'estime qu'on pourrait plutôt lui attribuer le statut 'I' puisqu'il est considéré comme inactif.

Permis de conduire

Le permis de conduire constitue également un attribut caractérisant un individu. Cet attribut indique si l'individu a obtenu son permis de conduire ou non. La variable de type « caractère » qui représente cet attribut vaut :

- 'O' : aucun permis de conduire
- 'P' : permis de conduire

À sa naissance, un individu n'a évidemment pas en sa possession un permis de conduire. C'est pourquoi, cet attribut vaut 'X', tout comme pour le statut professionnel. Néanmoins, il m'aurait

semblé plutôt logique de lui attribuer le type '0' puisqu'on sait qu'il n'a pas de permis de conduire.

Position dans le ménage

Parmi les caractéristiques d'un individu se trouve également la position qu'il occupe dans le ménage auquel il appartient. Il existe 4 positions différentes représentées par des variables de type « caractère » qui prennent les valeurs :

- 'H' : chef de ménage
- 'M' : conjoint
- 'A' : adulte supplémentaire
- 'C' : enfant

Lorsqu'un bébé naît, on lui attribue le type 'C' puisqu'il s'agit bien entendu d'un enfant. La position de l'individu au sein du ménage est calculée par le programme de génération de la population synthétique sur base du type du ménage, du nombre d'enfants dans le ménage, ainsi que sur base de la position de l'individu dans la liste des individus qui composent le ménage.

En effet, si l'individu se trouve en tête de la liste des individus qui composent le ménage, alors celui-ci est considéré comme le chef de ménage qui est caractérisé par la position 'H'. Prenons, par exemple, une famille composée d'individus ayant les identifiants 1, 2 et 3. L'individu 1 aura la position 'H' et l'individu 2 aura la position 'M'. L'individu 3 aura la position 'C' si le nombre d'enfants est égal à 1, sinon il aura la position 'A'.

Chaîne d'activités

Finalement, un individu est caractérisé par sa chaîne d'activités qui reflète son comportement tout au long d'une journée de 24 heures. Les 12 activités possibles sont reprises dans la table 1.1 à la section 1.3.2 du chapitre précédent.

Chaque activité est identifiée par des paramètres qui sont :

- le type
- le lieu
- la durée de l'activité
- la durée du trajet
- l'heure à laquelle se termine l'activité
- la distance parcourue

Dans VirtualBelgium, chaque individu peut construire un planning pour sa journée avec ses propres activités. Les valeurs de ces paramètres sont assignées tout au long de leur journée sur base des données collectées ou en fonction d'autres paramètres.

Maintenant que les caractéristiques des individus ont été développées, nous pouvons passer à celles concernant les ménages.

2.1.2 Ménages

Chaque individu de la population synthétique appartient à un ménage qui est également caractérisé par une série d'attributs. Ces derniers ainsi que leurs valeurs sont expliqués ci-dessous.

Identifiant

Tout comme les individus, chaque ménage possède un identifiant unique qui le caractérise. Celui-ci fait également partie des caractéristiques concernant les individus, comme expliqué à la section 2.1.1.

Le ménage possède également, comme attribut, un vecteur qui contient la liste des identifiants des individus qui appartiennent à ce ménage. Le premier individu de la liste est toujours le chef de ménage.

Lieu de domicile

Le ménage est caractérisé par son lieu de domicile qui est le même que celui de chacun de ses membres. Celui-ci est représenté par un code INS ainsi que par un noeud du réseau correspondant à sa localisation. Ces deux attributs sont totalement identiques à ceux caractérisant les individus.

Composition

Concernant la composition du ménage, ce dernier possède comme attribut le nombre d'enfants ainsi que le nombre d'adultes supplémentaires qui appartiennent à ce ménage. Ces deux attributs sont représentés par des nombres entiers qui peuvent prendre les valeurs suivantes :

- 0 à 5 : nombre d'enfants
- 0 à 2 : nombre d'adultes supplémentaires

Type

Enfin, une des caractéristiques les plus importantes concernant les ménages est le type de ce ménage. Dans VirtualBelgium, il existe 6 types de ménage représentés par des variables de type « chaîne de caractères » qui peuvent avoir les valeurs suivantes :

- "IF" : femme isolée (célibataire)
- "IH" : homme isolé (célibataire)
- "W" : famille monoparentale (mère et enfant(s))
- "M" : famille monoparentale (père et enfant(s))
- "C" : couple (sans enfant)
- "F" : famille (avec enfant(s))

Ces types peuvent être regroupés en 4 types de ménage différents : 'I' (isolé), 'N' (monoparental), 'C' (couple) et 'F' (famille). Ceux-ci seront utilisés lors de l'implémentation du modèle de migration résidentielle. Nous devons préciser que dans le cas où le ménage est de type 'C' ou 'F', il peut également contenir des adultes supplémentaires caractérisés par la position 'A' dans

le ménage.

Notons également que lorsqu'il s'agit d'un ménage de type 'F' ou 'N', autrement dit d'une famille classique ou monoparentale, le nombre d'enfants est toujours strictement supérieur à 0. Il y a toujours au moins un enfant, sinon on ne pourrait plus qualifier ce ménage de « famille ».

2.2 Processus de l'évolution temporelle

La population synthétique est générée par VirtualBelgium pour une année de référence, l'année 2001, sur base des données disponibles qui ont pu être collectées. Le but étant d'utiliser cet outil afin de réaliser des prospectives et prédictions pour le futur, il faut mettre en place un mécanisme qui fera évoluer temporellement la population synthétique de base.

L'évolution des attributs d'un individu ou d'un ménage peut être influencée par de nombreux facteurs tels que des événements (naissance, décès, mariage, déménagement, ...) ou d'autres attributs. En effet, le type d'un ménage peut changer l'année suivante et passer de couple sans enfant à couple avec enfant, suite à une naissance. Il peut également évoluer en fonction de l'âge du chef de ménage qui risque de décéder si son âge est trop élevé. Par exemple, le permis de conduire peut être obtenu en fonction de l'âge, du genre et du diplôme de l'individu.

VirtualBelgium utilise certaines techniques afin de modéliser l'évolution au cours du temps des attributs qui caractérisent les individus et les ménages de la population synthétique. En effet, plusieurs processus d'évolution d'une année à la suivante existent déjà et nous allons les décrire.

2.2.1 Évolution de l'âge

Le premier processus est totalement indispensable et consiste à faire évoluer l'âge des individus ainsi que la classe d'âges correspondante, au fil des années. Pour ce faire, il faut d'abord vérifier que l'individu ne va pas mourir entre l'année actuelle et l'année suivante, puisque dans ce cas, il ne servira à rien de faire évoluer son âge. Ce processus est expliqué à la section 2.2.3.

2.2.2 Naissance

Le second processus concerne les naissances. Tout d'abord, on vérifie si l'individu est de genre féminin, si son âge est compris entre 15 et 49 ans et si sa position dans le ménage est chef de ménage ou conjoint. Toutes ces caractéristiques feront de cet individu une mère potentielle. Ensuite, on évalue si la femme peut donner naissance sur base de taux de fécondité qui ont été mesurés pour l'année de référence, à savoir 2001, et qui dépendent de l'âge de la femme.

Si la femme donne naissance, un nouvel agent est créé pour représenter le nouveau-né. Ce dernier possède toutes les caractéristiques définies à la section 2.1.1. Il est ensuite ajouté à la liste des membres du ménage. Finalement, le type du ménage est mis à jour puisqu'une naissance

s'est produite et cela a pu affecter la composition du ménage. Par exemple, un couple sans enfant représenté par le type 'C', deviendrait alors une famille avec un enfant qui correspond au type 'F'.

2.2.3 Décès

Dès le début de l'évolution temporelle de la population synthétique de VirtualBelgium, il faut vérifier si l'individu risque de mourir ou non. Pour y parvenir, on utilise des taux de mortalité qui ont été mesurés pour l'année de référence et qui dépendent du genre et de l'âge de l'individu.

Si l'individu est vivant, on peut exécuter les deux processus expliqués ci-dessus. Dans le cas contraire, l'individu est décédé et il faut donc le retirer de la simulation en supprimant son identifiant de la liste des individus qui composent son ménage. Il faut également veiller à ce que le ménage de cet individu ne soit pas vide après la disparition de ce dernier. Si c'est le cas, il faut retirer le ménage de la simulation et de la liste des agents. Sinon, il suffit de mettre à jour le type du ménage après la perte d'un de ses membres. Dans ce cas, la position de l'individu au sein du ménage est également mise à jour pour chacun des membres du ménage. Par exemple, dans le cas où le chef de ménage décède, le conjoint devient le nouveau chef de ménage.

2.2.4 Autres processus

Il existe également d'autres processus d'évolution temporelle qui ne sont pas encore implémentés dans le programme de simulation de la population synthétique. En effet, pour le moment les mariages et les divorces ne font pas partie des processus d'évolution de VirtualBelgium. On ne sait donc pas si les ménages de type 'C' sont composés de couples mariés ou non mariés. Il en va de même pour les ménages de type 'F' qui représentent des familles. Ces phénomènes de mariage et de divorce sont très importants car lorsqu'un couple se forme et se marie ou lorsque deux personnes divorcent, il y a de fortes chances pour que cela engendre une modification dans les ménages et surtout un déménagement. Ce processus de migration résidentielle n'est pas encore implémenté mais cela est expliqué dans la section suivante.

Certaines caractéristiques des individus n'évoluent pas non plus pour le moment. En effet, lorsque les individus de la population synthétique sont générés la première année, on leur associe des attributs tels que la prise de possession du permis de conduire, le plus haut diplôme obtenu ou encore le statut professionnel. Cependant, l'évolution au fil des années de ces attributs n'est pas encore implémentée. Il faudrait que le diplôme de l'individu évolue en fonction de son âge et des écoles dans lesquelles il étudie. L'individu devrait également obtenir son permis de conduire à partir d'un certain âge et en fonction d'autres paramètres. Concernant le statut professionnel, un enfant qui possède le statut étudiant devrait passer en statut actif lorsqu'il devient adulte et en fonction d'autres paramètres à nouveau.

2.3 Migration résidentielle

Pour l'instant, VirtualBelgium ne considère pas les migrations résidentielles, ce qui peut engendrer des soucis. En effet, lorsque nous faisons évoluer la population synthétique, les enfants grandissent et sont supposés quitter le ménage et la maison familiale afin de créer leur propre ménage. Ce phénomène n'est pas encore implémenté dans le programme. De plus, un individu peut déménager suite à un mariage ou un divorce, à un changement de travail, à une naissance, ... Pour prendre en compte tous ces phénomènes, nous allons implémenter un modèle de migration résidentielle qui va prédire le choix de migration d'un individu en fonction de ses caractéristiques et de celles du ménage auquel il appartient.

La population synthétique de VirtualBelgium et son évolution vont nous permettre d'utiliser le modèle de migration résidentielle dans le but de réaliser des prévisions à propos de la population belge. Afin de savoir les caractéristiques qui seront utiles à cette modélisation, nous nous sommes basés sur le projet MOBLOC¹ qui avait pour objectif de comprendre le lien entre la migration journalière et la migration résidentielle. Celui-ci, ainsi que les modèles créés pour atteindre son but sont expliqués dans le chapitre suivant.

1. Mobilities and Long Term Location Choices in Belgium

Chapitre 3

MOBLOC : présentation des modèles

La mobilité journalière et la migration des ménages possèdent de nombreuses interactions. En effet, un changement de résidence induit un changement de comportement dans la mobilité et peut être induit par des difficultés de mobilité journalière comme, par exemple, suite à un nouveau travail. Le projet MOBLOC a étudié le choix de résidence fait par les ménages, la demande de transport et l'évolution de l'accessibilité qui en résulte. Les modèles développés dans ce projet seront expliqués dans ce chapitre sur base des deux rapports publiés [2, 7].

3.1 Objectif

Les choix de lieu de résidence se sont élargis suite à l'évolution considérable des systèmes de transport. Les gens ont tendance à s'installer de plus en plus loin des centres d'activités urbains, poussés par le phénomène d'expansion urbaine. L'usage de la voiture est de plus en plus prisé pour les voyages quotidiens et amène de nouveaux comportements de mobilité.

Un des objectifs principaux de MOBLOC a été de montrer le lien entre la mobilité journalière et la migration résidentielle. Ce projet a aussi cherché à connaître les facteurs qui influencent un ménage et qui le poussent à déménager, ainsi que les critères sur lesquels le ménage se base afin de choisir une nouvelle commune de résidence. Cette tendance à migrer peut varier en fonction de certaines caractéristiques individuelles qui ont été identifiées par MOBLOC.

3.2 Méthodologie

MOBLOC a construit toute une série de modèles centrés autour de la migration résidentielle. Les liens entre ces différents modèles ainsi que les méthodes de modélisation utilisées pour les construire, sont expliqués tout au long de ce chapitre.

Sur la figure 3.1, nous pouvons voir les différents modèles nécessaires ainsi que les données utilisées en entrées et en sorties entre ces modèles. Étant donné l'objectif de MOBLOC, les modèles principaux sont celui de migration résidentielle et celui d'accessibilité.

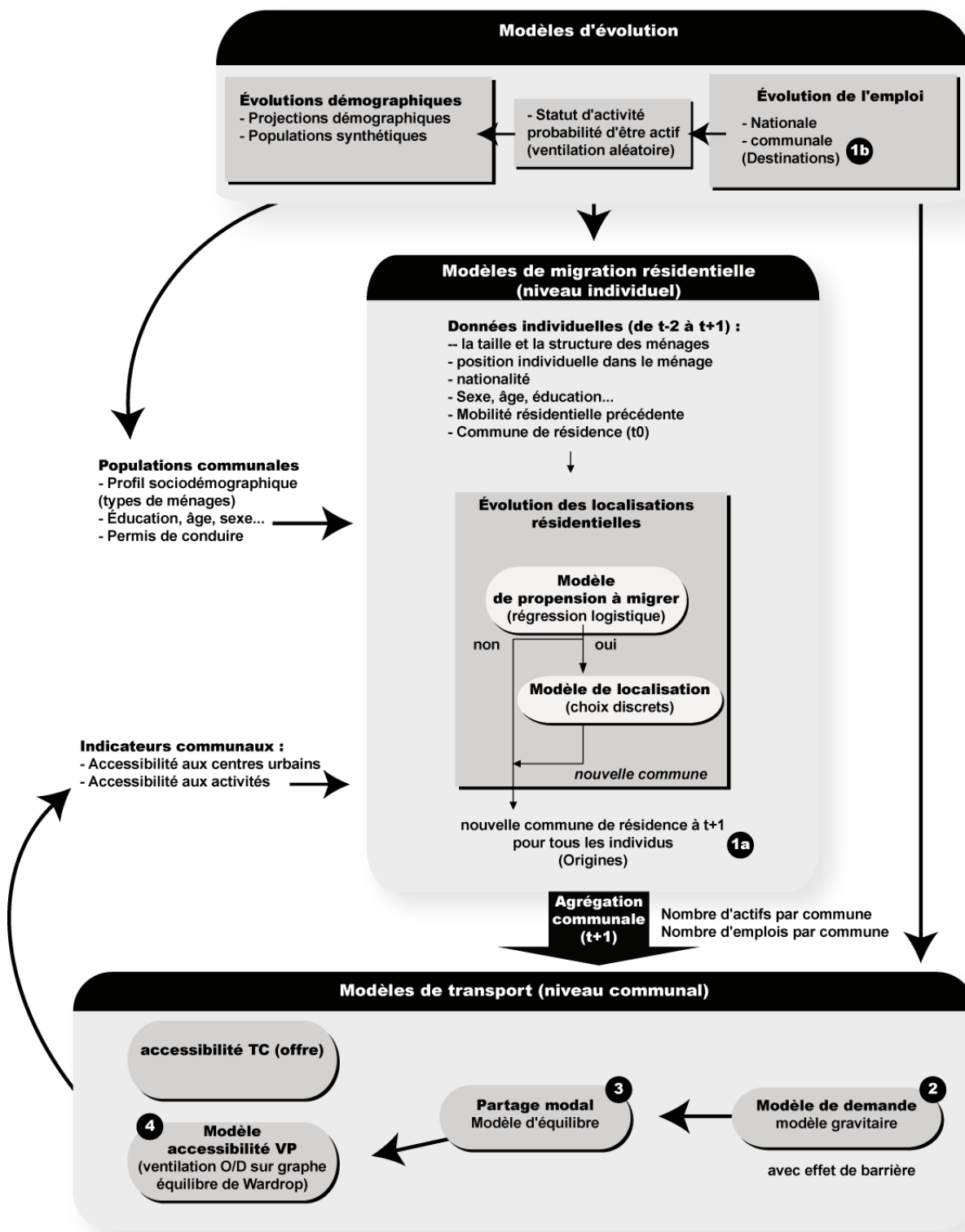


FIGURE 3.1 – Schéma du projet MOBLOC [8]

Commençons par le modèle principal qui nous intéresse dans le cadre de ce travail : le modèle de migration résidentielle (1a). Celui-ci prend comme entrées des données individuelles telles que l'âge, le genre, le niveau d'éducation, la taille et le type du ménage, la position dans le ménage,

... La sortie fournie par ce modèle est la nouvelle commune de résidence de l'individu pour l'année suivante. Ce modèle est constitué de deux sous-modèles. Le premier, appelé modèle de propension à migrer, a pour but de prédire si un déménagement va survenir entre l'année actuelle et la suivante. Il indique donc si un individu va changer de commune de résidence, grâce à une régression logistique binaire. Notons que MOBLOC se limite aux déménagements entre communes et néglige ceux au sein d'une même commune. Le deuxième, appelé modèle de localisation, simule le choix de résidence parmi les 588 autres communes belges, si le premier modèle indique que l'individu va déménager. La technique utilisée est celle du choix discret [27].

Concernant les modèles de transport, nous avons en premier lieu le modèle de demande (2) qui nécessite en entrée, le nombre de personnes actives et le nombre d'emplois par commune, qui sont donnés respectivement par le modèle de migration (1a) et le modèle d'évolution (1b). Ensuite, on fait appel à un modèle de répartition modale (3) qui a pour but d'extraire uniquement les déplacements effectués en voiture afin de les fournir au modèle d'accessibilité en véhicule privé (4). Ce dernier modèle permet d'obtenir une matrice des temps de parcours entre communes qui sont utilisés dans le calcul d'indices d'accessibilité aux emplois et aux services par commune. Ces indices sont ensuite utilisés par le modèle de localisation pour mesurer l'attractivité de ces communes.

Enfin, dans le but de réaliser une prospective, les données prises en entrées du modèle de migration résidentielle doivent être mises à jour. Cela est possible grâce aux modèles d'évolution (1b) qui utilisent les populations synthétiques.

Dans les sections suivantes, nous allons expliquer avec plus de détails ces différents modèles qui constituent le projet MOBLOC.

3.3 Modèles de transport

Les modèles de transport au niveau des communes sont répartis en trois modèles : un modèle de demande, un modèle de répartition modale et un modèle d'accessibilité. Ceux-ci ont pour objectif final de fournir des indicateurs d'accessibilité qui influenceront le choix de la commune de résidence lors d'un déménagement. Ces 3 modèles sont décrits ci-dessous séparément.

3.3.1 Modèle de demande en mobilité

Le premier modèle de transport est le modèle de demande en mobilité. Celui-ci va contribuer à modéliser l'accessibilité entre chaque commune au fil du temps, via la simulation des trajets pour se rendre au travail réalisés entre toutes les paires origine-destination possibles, ce qui représente 589×589 communes belges.

3.3.1.1 Données

Les données nécessaires à la première itération du modèle sont la matrice origine-destination¹ complète fournie par l'enquête socio-économique INS de 2001 et la matrice des distances euclidiennes entre toutes les paires de communes. Pour les autres itérations, nous n'avons plus besoin de toute la matrice origine-destination mais uniquement des marges de celle-ci et de la matrice des distances. Ces marges représentent le nombre d'emplois par commune et le nombre de personnes actives par commune, qui sont fournis respectivement par le modèle d'évolution des emplois et le modèle de localisation.

3.3.1.2 Méthodologie

La première étape du modèle consiste à calculer les marges de la matrice origine-destination grâce aux données fournies. La deuxième étape consiste à représenter cette demande en mobilité en utilisant un modèle gravitaire² avec une double contrainte, afin de rester cohérent avec les autres modèles de MOBLOC. En effet, nous devons tenir compte de deux contraintes pour le modèle gravitaire :

- le nombre de personnes qui vont au travail dans une commune ne peut pas excéder le nombre d'emplois disponibles dans cette commune.
- le nombre de navetteurs dans une commune ne peut pas excéder le nombre d'employés qui habitent dans cette commune.

Le modèle gravitaire va servir à construire les cellules internes de la matrice origine-destination qui prend uniquement en compte la demande en mobilité entre le travail et la maison. Il devra tenir compte de la diagonale de la matrice qui représente le flux au sein d'une même commune et des effets tels que la séparation linguistique en Belgique et la forte concentration présente autour de Bruxelles, qui affectent l'effet de la distance sur les interactions spatiales. En effet, l'interaction entre deux points décline lorsque la distance entre eux augmente.

3.3.1.3 Modèle gravitaire

Le modèle gravitaire va réaliser la modélisation des flux origine-destination concernant les trajets pour se rendre au travail. Pour ce faire, il va calculer les cellules de la matrice des flux origine-destination.

1. Matrice carrée à deux entrées, les communes correspondant aux lieux de résidence et les communes correspondant aux lieux de travail, dont les éléments représentent le nombre de déplacements entre deux communes.

2. Le modèle gravitaire a été développé en 1931 par William J. Reilly. Il permet de connaître l'attraction relative de deux centres urbains sur une population localisée dans une zone intermédiaire. Selon la loi de Reilly, l'attraction est proportionnelle à la taille des centres et inversement proportionnelle au carré des distances entre la zone intermédiaire et les centres urbains concernés.[17]

	1	j	n	O_i
1	F_{ij}			
i				
n				
D_j				T

FIGURE 3.2 – Matrice des flux origine-destination

Cette matrice est représentée à la figure 3.2, où

- F_{ij} est le flux entre les communes i et j
- O_i est le nombre d'employés qui habitent dans la commune i
- D_j est le nombre de personnes qui vont travailler dans la commune j
- T est le total employés/emplois

Comme il s'agit d'un modèle gravitaire, on suppose que l'intensité des flux entre les communes i et j dépend des poids que représentent le nombre d'emplois et le nombre d'employés par commune, et qu'elle est inversement proportionnelle à la distance entre elles. Cela signifie donc que plus la distance diminue et les communes sont proches, plus l'intensité du flux sera forte entre ces deux communes. En effet, il paraît plutôt logique d'avoir un nombre plus élevé de déplacements du domicile au travail entre deux communes qui sont voisines, par rapport à deux communes très éloignées.

Nous pouvons formaliser le modèle gravitaire de la manière suivante :

$$F_{ij} = A_i B_j O_i D_j f(d_{ij}, \beta)$$

où f est une exponentielle, une fonction de Pareto ou une combinaison des deux qui dépend de deux paramètres : d_{ij} , la distance entre les communes i et j , et le paramètre β qui représente la friction de la distance³. Cette fonction s'exprime sous la forme :

$$f(d_{ij}, \beta) = \exp^{\beta * d_{ij}} \text{ ou } d_{ij}^{\beta} \text{ ou } \exp^{\beta_1 * d_{ij}} * d_{ij}^{\beta_2}$$

Les paramètres A_i et B_j sont des contraintes sur les marges qui doivent être estimées simultanément avec

$$\forall i : A_i = \frac{1}{\sum_j B_j D_j f(d_{ij}, \beta)}$$

3. La friction ou frottement de la distance décrit la force qui crée le déclin de la distance. À cause de cette friction, les interactions spatiales auront tendance à avoir lieu plus souvent sur de plus courtes distances.[29]

$$\forall j : B_j = \frac{1}{\sum_i A_i O_i f(d_{ij}, \beta)}$$

où $O_i = F_i$ et $D_j = F_j$.

Afin d'améliorer la qualité du modèle, il faudrait tenir compte de la concentration élevée des emplois dans la région de Bruxelles pour obtenir une estimation des flux proche de la réalité. Ceci peut être réalisé grâce à un effet de barrière entre régions. En effet, étant donné que nous travaillons sur la population de la Belgique, il faut tenir compte des régions et des barrières linguistiques. Un moyen d'y parvenir consiste à introduire de nouveaux paramètres dans le modèle qui modélisent cet effet de frontière. Puisque la Belgique contient trois régions, la formulation du modèle devient :

$$F_{ij} = A_i B_j O_i D_j f(d_{ij}, \beta, \gamma)$$

avec

$$f(d_{ij}, \beta, \gamma) = \exp^{\beta_1 * d_{ij}} * d_{ij}^{\beta_2} * \gamma_1^{FB} * \gamma_2^{FW} * \gamma_3^{BW}$$

où FB , FW et BW sont des variables binaires qui représentent les flux entre régions. Elles valent 1 si le flux passe la frontière respectivement entre la Flandre et Bruxelles, la Flandre et la Wallonie, Bruxelles et la Wallonie, et elles valent 0 dans le cas contraire. On peut mesurer si les flux d'une région vers une autre vont freiner ou accélérer grâce aux paramètres de barrière, notés γ . Si $\gamma < 1$, les flux auront tendance à diminuer lors du passage entre régions et inversement si $\gamma > 1$, on observera une intensification des flux.

Le processus itératif afin d'estimer les paramètres A_i et B_j est détaillé à la figure 3.3. Grâce à la matrice origine-destination fournie en entrée du modèle, on peut réaliser la première itération du processus en commençant par l'étape 1. Une fois que B_j et le compteur d'itérations sont initialisés, on passe à l'itération suivante avec les étapes 2 et 3 qui constituent le calcul de A_i et de B_j . Ensuite, à l'étape 4, on calcule la différence entre A_i à l'itération actuelle et à l'itération précédente afin de voir si le paramètre converge vers une valeur. On fait de même pour le paramètre B_j . Si les deux paramètres convergent, on arrête la procédure itérative et on passe à l'étape 5. Dans le cas contraire, on recalcule ces paramètres en recommençant à l'étape 2 jusqu'à ce qu'ils convergent. L'étape finale consiste à calculer le flux du trajet pour se rendre au travail de la commune i vers la commune j .

3.3.2 Modèle de répartition modale (Modal split)

Le deuxième modèle faisant partie des modèles de transport est un modèle de répartition modale, appelé modal split. « Le partage modal permet de distinguer la proportion d'utilisateurs des différents modes de transport » [26]. Cette répartition des flux pour les trajets du domicile vers le travail est effectuée pour les 2 modes de transport suivants : les véhicules privés et les transports en commun.

Ce modèle prend comme entrée, la matrice origine-destination qui concerne tous les modes

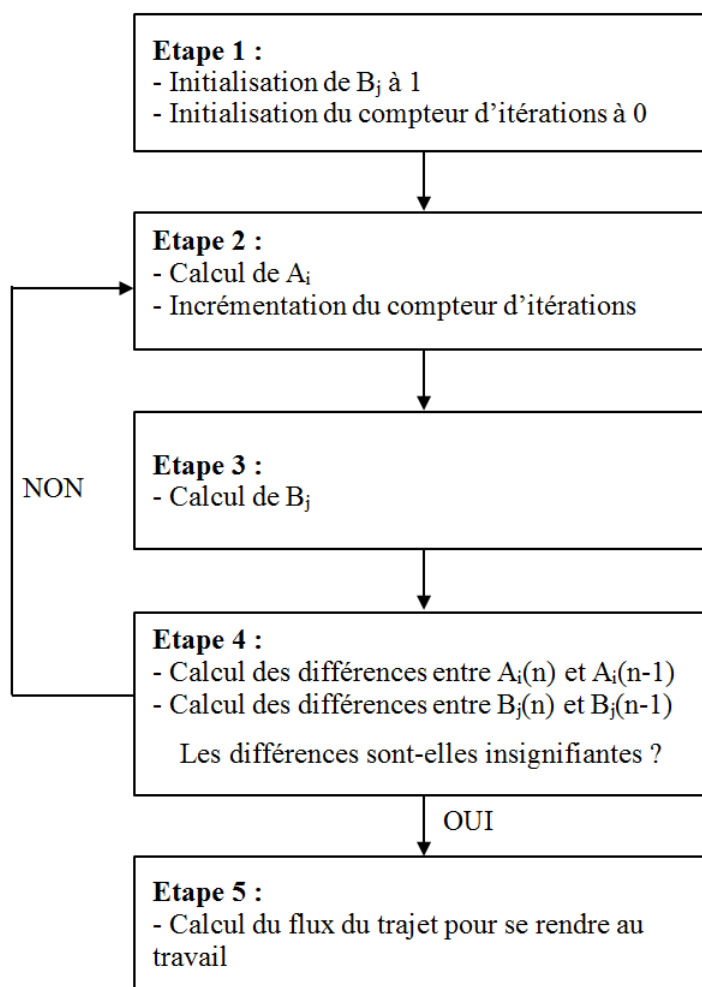


FIGURE 3.3 – Processus itératif du modèle gravitaire avec une double contrainte

de transport et qui est fournie par le modèle de demande en mobilité. Son objectif est d'extraire de cette matrice uniquement les déplacements effectués en voiture grâce au partage modal et de créer ainsi une nouvelle matrice qui contiendra seulement les demandes en mobilité concernant la voiture. Cette seconde matrice sera utilisée par le modèle d'accessibilité pour la voiture décrit dans la section suivante, dans le but de calculer des indicateurs d'accessibilité.

Le partage modal peut être appliqué sur deux éléments :

- soit sur les **cellules** de la matrice origine-destination qui représentent les flux entre les communes d'origine et de destination,
- soit sur les **marges** de la matrice origine-destination qui représentent le nombre de personnes actives qui habitent dans les communes d'origine et le nombre de personnes qui vont travailler dans les communes de destination.

Comme le choix modal⁴ dépend de l'origine, mais également de la destination, on choisit d'appliquer le partage modal sur les cellules de la matrice origine-destination, puisque les marges ne

4. Étude du mode de transport utilisé.

dépendent que de l'origine ou de la destination, mais pas des deux à la fois. Le choix modal des cellules qui valent zéro est toutefois conservé, car cela correspond à des paires origine-destination que personne n'utilise pour se rendre au travail et il faut conserver cette information.

Ce modèle est très important, car l'assignation du trafic pourrait entraîner une saturation du réseau dans le cas où le nombre de conducteurs est beaucoup trop élevé, suite à une mauvaise estimation trop à la hausse.

3.3.3 Modèle d'accessibilité

Le dernier modèle de transport est un modèle d'accessibilité. Celui-ci est divisé en un modèle d'accessibilité pour la voiture et un autre indépendant du premier pour les transports en commun. L'accessibilité, dans notre cas, est mesurée en terme de temps de parcours d'un point à un autre point. Ceux-ci sont calculés sur base des plus courts chemins selon deux modes : la voiture privée et les transports en commun (train, bus, tram ou métro). L'accessibilité est inversement proportionnelle à la distance entre la commune d'origine et la commune de destination.

L'objectif de ce modèle consiste à fournir une mesure de l'accessibilité entre chaque paire de communes, ainsi qu'à étudier les interactions entre les tendances démographiques et la mobilité. Ce modèle va jouer un rôle important dans le choix du lieu de résidence des individus et plus particulièrement dans celui des personnes actives. En effet, le temps de parcours requis entre le travail et le lieu de résidence peut influencer le choix de ce dernier.

Les nouvelles tendances concernant le trafic et la demande en mobilité au niveau communal pourront être analysées grâce à ce modèle. En effet, la congestion du trafic pourrait s'amplifier suite à l'extension de la périurbanisation, ce qui entraînerait une augmentation des temps de parcours et une perte pour l'attractivité des régions rurales.

3.3.3.1 Modèle pour la voiture

L'entrée du modèle d'accessibilité pour la voiture est la matrice de la demande en mobilité, représentée sous la forme d'une matrice origine-destination d'étudiants et de personnes actives qui utilisent la voiture comme principal moyen de transport pour se rendre de la maison à l'école ou au travail. Ce modèle utilise également le réseau routier belge et ses paramètres.

Ces données proviennent de l'enquête socio-économique INS de 2001 et la matrice origine-destination pour les personnes utilisant la voiture est fournie par le modèle de répartition modale. Remarquons que la diagonale de la matrice fournie, contenant le nombre de déplacements des personnes qui vivent et travaillent dans une même commune, ne sera pas prise en considération.

Les différentes étapes de construction de ce modèle sont les suivantes :

1. Choix des points représentatifs

Tout d’abord, compte tenu de la densité du réseau de transport belge, on a choisit de simplifier le modèle et de ne pas considérer toutes les localisations. Étant donné que la modélisation est réalisée sur la Belgique, la première étape consiste à choisir un point représentatif pour chacune des 589 communes belges en prenant un unique centre géographique par commune. Dans ce cas, l’accessibilité de ce point sera assimilée à celle de toute la commune.

Néanmoins, le choix d’un point unique représentant la commune n’est pas toujours évident à réaliser étant donné que certaines communes sont issues de la fusion d’autres communes (1977). D’après Van der Haegen [28], on distingue quatre types de communes : des agglomérations, des banlieues, des zones résidentielles de migrants alternants et d’autres communes non polarisées. Pour les agglomérations, on prend le centre-ville comme point représentatif, tandis que pour les autres types de communes, on se base sur le géocodage du nom de la commune réalisé grâce à Google Maps. Il s’agit souvent de l’agglomération principale de la commune.

2. Représentation du réseau routier

Le réseau routier belge utilisé contient des autoroutes et des routes nationales de niveaux 1, 2 et 3. Dans le cas d’une modélisation intercommunale, c’est généralement suffisant comme on peut le voir à la figure 3.4. On remarque que la capitale ainsi que les grandes villes de Belgique sont bien reliées par le réseau. La seconde étape consiste à effectuer une simplification de ce réseau routier. Ce dernier est représenté sous la forme d’un graphe orienté $G = (V, E)$, où V est l’ensemble des sommets qui représentent les intersections du réseau et E est l’ensemble des arcs qui représentent les portions de route du réseau.

Les sommets correspondent aux points représentatifs des communes et ils doivent être reliés par des connecteurs qui correspondent aux arcs. Si un point représentant une commune n’apparaît pas sur le réseau, on crée des routes « virtuelles » afin de relier ces points au réseau. Chaque arc est caractérisé par une longueur, une vitesse de circulation en flux libre et une capacité routière.

L’étape suivante consiste à fixer les paramètres du modèle d’assignation du trafic expliqué ci-dessous. Les vitesses en flux libre sont fixées en fonction du nombre de voies de circulation, de la présence ou non d’une bande centrale, ainsi que du type d’urbanisation traversée par cette route. La capacité routière correspond, quant à elle, à 2000 unités de voiture particulière (UVP) par heure et par bande de circulation. En milieu urbain, elle vaut 1200 UVP par heure et par bande de circulation, tandis que hors agglomération, elle varie entre 1400 et 2000 UVP par heure et par bande de circulation.

3. Modèle d’affectation de trafic

À présent, on peut construire deux modèles d’affectation de trafic : l’un concernant les

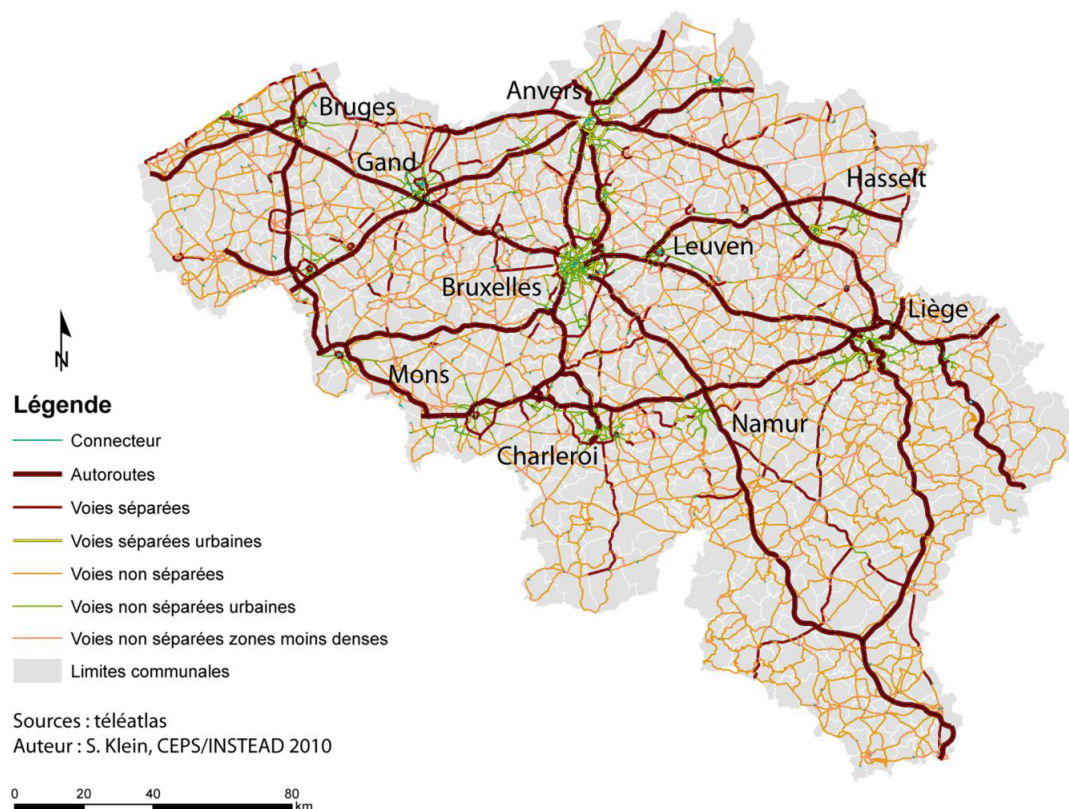


FIGURE 3.4 – Réseau routier de Belgique [7]

périodes en heures creuses et l'autre la période de pointe du matin entre 7h et 8h.

Pour le modèle en heures creuses, on suppose que les trajets s'effectuent sur les tronçons à la vitesse en flux libre. Afin de déterminer le chemin qui possède le coût minimal, c'est-à-dire dans ce cas-ci le temps de parcours minimal, on applique l'algorithme du plus court chemin de Dijkstra⁵. Grâce à celui-ci, on construit la matrice des temps de parcours entre chaque paire de communes.

Pour l'affectation en heures de pointe, le modèle choisi se base sur la première hypothèse de Wardrop [11] qui spécifie les conditions afin d'atteindre un équilibre utilisateur. Ce dernier signifie qu'aucun usager ne peut améliorer son temps de parcours en modifiant son itinéraire. Cela est possible si toutes les routes utilisées sur le réseau pour aller d'un point à un autre possèdent le même temps de parcours et si ce dernier est inférieur à celui des routes non utilisées. Pour ce faire, on affecte une nouvelle vitesse à chaque portion de route dans le réseau afin que le temps de parcours soit équivalent pour toutes les routes utilisées entre ces deux points. Cette étape est réalisée afin de tenir compte des embouteillages dans le calcul du temps de parcours. On peut alors calculer la matrice des temps de parcours entre chaque commune grâce à l'algorithme de Dijkstra.

5. En théorie des graphes, l'algorithme de Dijkstra permet de déterminer le plus court chemin entre deux sommets d'un graphe connexe pondéré dont le poids lié aux arêtes est positif ou nul. Cette définition est tirée de la référence [18] où on peut trouver les étapes de l'algorithme ainsi qu'un exemple.

4. Calcul d'accessibilité

Finalement, on a construit une matrice d'accessibilité entre chaque paire de communes qui contient dans chacune de ses cellules le temps de parcours du meilleur itinéraire en voiture entre deux communes. Ces temps de parcours sont utilisés afin d'obtenir des indicateurs d'accessibilité (pour l'emploi et les services) communaux qui serviront à modéliser l'impact de la mobilité journalière sur les choix résidentiels. En effet, ces indicateurs feront partie des entrées du modèle de localisation, expliqué ci-dessous, pour mesurer l'attractivité d'une commune. On suppose que le trajet domicile-travail influencera le choix résidentiel des personnes actives, principalement concernant le temps d'accès aux heures de pointe.

3.3.3.2 Modèle pour les transports en commun

Un modèle d'accessibilité pour les transports en commun a été prévu afin de calculer les temps de parcours effectués en transports en commun entre chaque paire de communes. Les données nécessaires pour ce modèle sont : les routes, les arrêts et les horaires des transports en commun en Belgique. Certaines données ont été fournies par la SRWT⁶. Cependant, ce modèle n'a pas pu être construit par manque de données complètes concernant les horaires des transports en commun.

Comme le temps de parcours est calculé sur base des horaires fournis et du service proposé, il n'aurait pas été lié à la demande en mobilité. De plus, le modèle n'aurait pas pu être mis à jour à chaque étape et donc l'accessibilité n'aurait pu être calculée qu'une seule fois pour toute la période de simulation. En effet, on ne peut pas prévoir exactement où se situeront les routes créées dans le futur, et on ne sait pas comment va évoluer le service proposé concernant les transports en commun pour les années suivantes.

3.4 Modèle de migration résidentielle

Le modèle de migration résidentielle, qui constitue l'une des briques les plus importantes de MOBLOC, décrit le comportement d'un individu par rapport à la migration, c'est-à-dire à un changement de commune de résidence entre le 1^{er} janvier d'une année et le 1^{er} janvier de l'année suivante. Son objectif est de prédire quelle sera, l'année suivante, la commune de résidence de chaque individu.

Les données utilisées afin de modéliser la migration résidentielle proviennent d'une base de données obtenue grâce au Registre National belge et décrivant des « états » au niveau individuel à chaque 1^{er} janvier de l'année 2001 jusqu'à l'année 2006. Des données de recensement national provenant de l'enquête socio-économique ESE de 2001, constituent une autre source de données

6. Société Régionale Wallonne de Transports

pour construire ce modèle. Cependant, suite à un manque de données, nous ne pouvons malheureusement pas observer les migrations réelles qui se produisent parfois au sein d'une même commune. MOBLOC a donc uniquement considéré les migrations entre communes différentes et qui se produisent entre deux 1^{er} janvier successifs, puisque le Registre National ne nous donne que ces informations.

On peut subdiviser ce modèle en deux sous-modèles : un modèle de propension à migrer ainsi qu'un modèle de localisation. Ces modèles seront détaillés dans les sections suivantes.

3.4.1 Modèle de propension à migrer

Depuis 1988, le phénomène de migration résidentielle n'a cessé d'augmenter, passant de 379.000 individus belges en 1988 à 481.000 en 2004 et même 532.000 en 2009⁷. On remarque également que la propension à migrer a augmenté à tous les âges de 0 à 65 ans. De plus, on observe que le phénomène de périurbanisation s'étend sur le territoire belge. En effet, les jeunes ménages recherchent des terrains à bâtir ainsi que des logements moins cher et ils doivent donc s'éloigner des villes afin de les trouver.

On observe également de plus en plus de transformations dans le ménage, ce qui entraîne une croissance de la migration résidentielle. En effet, un ménage va vouloir chercher une nouvelle résidence lors de situations telles que :

- un nouveau couple se forme
- un couple se sépare
- un adulte du ménage trouve un nouvel emploi loin de son lieu de résidence
- un étudiant ou un adulte retraité décide de déménager
- la résidence est trop petite ou trop grande pour le ménage, suite à une naissance, un décès ou encore un mariage.

Ce modèle prédit si une migration d'une commune à une autre va se produire entre l'année actuelle et la suivante. Autrement dit, il prédit si un individu va déménager de sa commune. Si la réponse est oui, le modèle de localisation simulera le choix de localisation entre les 588 autres communes belges.

3.4.1.1 Sélection des variables

Les variables explicatives utilisées pour construire ce modèle ont été sélectionnées sur base de publications [10, 15] et sur base des données collectées. Celles-ci comportent des caractéristiques individuelles et des caractéristiques concernant le logement. La variable à expliquer est une variable binaire qui dit si l'individu va déménager. Si c'est le cas, la variable vaudra 1, sinon

7. Ces statistiques proviennent de Statbel, SPF Économie, PME, Classes moyennes et Énergie, Direction générale Statistique et Information économique.

elle vaudra 0.

Afin de savoir quelles variables explicatives garder dans le modèle, des tables de contingences ont été construites par MOBLOC. Ces dernières constituent un moyen de représenter les données en comptant l'effectif correspondant au croisement de deux caractéristiques. Dans notre cas, on a dénombré le nombre d'individus qui ont déménagé en fonction de chaque valeur des variables explicatives choisies. La table 3.1 nous indique les catégories qui ont la plus élevée et la plus faible propension à migrer, sur base de ces tables de contingence.

VARIABLES	CATÉGORIES AYANT LA PLUS ÉLEVÉE PROPENSION À MIGRER	CATÉGORIES AYANT LA PLUS FAIBLE PROPENSION À MIGRER
Âge	19-29	55-74
Genre	Homme	Femme
Nationalité	Congolaise RDC	Turque
Type du ménage	Cohabitants sans enfant	Couple marié sans enfant
Taille du ménage	1 individu	4 individus
Position dans le ménage	Autre (que chef, conjoint et enfant)	Conjoint
Migration précédente	Oui	Non
Lieu de résidence	Agglomération (sans le centre-ville)	Petite ville
Niveau d'éducation	Éducation supérieure	Éducation primaire ou aucune
Statut professionnel	Demandeur d'emploi	Retraité
Type d'occupation du logement	Locataire dans le secteur privé	Propriétaire
Type de logement	Appartement	Maison à 4 façades

TABLE 3.1 – Catégories ayant la plus élevée et la plus faible propension à migrer en fonction des variables explicatives

Ce tableau nous montre que l'homme semble avoir une plus forte tendance à migrer que la femme, d'après la base de données fournie. Les individus plus jeunes qui ont entre 19 et 29 ans ont également plus de chance de migrer par rapport aux personnes plus âgées qui ont entre 55 et 74 ans. On remarque également des situations qui nous semblent cohérentes. En effet, un

couple marié ou un ménage composé d'un plus grand nombre d'individus auront moins tendance à déménager. Il en va de même pour des ménages qui sont propriétaires de leur logement, qui possèdent une maison à 4 façades et habitent dans des petites villes. Au contraire, un individu qui a effectué une migration l'année précédente a plus de chance de migrer à nouveau. C'est aussi le cas des demandeurs d'emploi qui risquent de déménager près de leur futur emploi, et des étudiants qui ont obtenu leur diplôme d'études supérieures, qui vont sans doute quitter la maison familiale.

Le but de ce modèle étant de se focaliser sur les caractéristiques individuelles afin de voir comment celles-ci influencent le choix de migrer, MOBLOC a choisi d'exclure le lieu de résidence ainsi que le type de logement. De plus, le niveau d'éducation a été préféré au statut professionnel car ceux-ci sont fort corrélés et le statut professionnel provenait parfois de données incohérentes. En effet, un étudiant pouvait avoir un statut actif.

3.4.1.2 Méthodologie

Pour la construction du modèle, MOBLOC a utilisé une régression logistique binaire qui sera expliquée au début du chapitre suivant. Il a dû faire face à deux contraintes : le pas annuel pour la migration et la dimension dynamique. En effet, comme on a les données à chaque 1^{er} janvier, on a dû intégrer une unité temporelle pour les variables avec un pas d'une année pour l'évolution. Ensuite, une dimension dynamique a été introduite pour les variables car les informations concernant le passé d'un individu peuvent aider à expliquer son comportement de migration selon s'il a déjà migré par le passé ou s'il y a eu une modification dans la structure du ménage. De plus, cette composante dynamique va aussi tenir compte des événements anticipés tels que des mariages ou des naissances qui sont prévus.

Pour ce faire, on a créé des variables de transition à partir des données du Registre National pour la taille et le type du ménage, la position dans le ménage, la nationalité et la migration précédente. Des variables synthétiques ont été créées pour représenter la taille et le type du ménage en une seule variable qui tient compte des deux. Lorsque deux variables d'état sont identiques pour deux années successives, la variable de transition correspondante sera équivalente à « pas de changement ».

La construction du modèle a été réalisée à partir d'un échantillon des données suite aux problèmes rencontrés par rapport à la taille très élevée de la base de données et au temps de calcul. Pour calibrer le modèle, MOBLOC a utilisé 70% de cet échantillon et les 30% restants ont servi à valider le modèle.

3.4.1.3 Modèle

Comme les données collectées commencent en 2001 et finissent en 2005, on a construit différents modèles qui servent à prédire la propension à migrer entre 2002 et 2003 et d'autres modèles avec d'autres variables entre 2003 et 2004. De cette manière, on peut inclure des variables de

transition pour les périodes 2001-2001, 2002-2003, 2003-2004 et 2004-2005.

La procédure stepwise [1] a été utilisée afin de sélectionner une par une les variables significatives afin de les inclure dans le modèle. Elle peut également retirer une variable par la suite, si elle n'est plus significative suite à l'ajout d'une autre variable.

Les modèles testés ont été comparés avec plusieurs critères et ont tous une bonne calibration. Cependant, sur bases de plusieurs tests statistiques décrits dans les rapports de MOBLOC, un modèle est apparu légèrement meilleur que les autres.

VARIABLES	NOMBRE DE MODALITÉS	MODALITÉS
Âge	6	0-18, 19-29, 30-44, 45-54, 55-74, 75 et plus
Genre	2	Homme, Femme
Nationalité	8	Belge, Congolaise, Marocaine, Turque, Europe des 12, Europe des 15, Europe de l'Ouest et pays non européens, Reste de l'Europe de l'Est
Type du ménage	8	Couple marié sans ou avec enfant, Couple cohabitant sans ou avec enfant, Isolé, Monoparental, Collectif ⁸ , Autre
Taille du ménage	5	1, 2, 3, 4, 5 ou plus individus
Position dans le ménage	4	Chef de ménage, Conjoint, Enfant, Autre
Migration précédente	2	Oui, Non
Niveau d'éducation	5	Aucun ou primaire, Secondaire inférieur, Secondaire supérieur, Éducation supérieure Aucune réponse
Type d'occupation du logement	5	Propriétaire, Locataire d'un logement privé, Locataire d'un logement social, Aucune réponse, Locataire d'un logement gratuit

TABLE 3.2 – Variables du modèle de propension à migrer et leurs modalités

Les variables sélectionnées pour ce modèle ainsi que leurs modalités sont reprises dans la table 3.2. Notons que les variables concernant le type et la taille du ménage ont été regroupées en une seule variable. De plus, les variables de transition incorporées à ce modèle sont : l'évolution du type du ménage entre l'année précédente et l'année actuelle, ainsi qu'entre l'année actuelle et l'année suivante, et l'évolution de la position dans le ménage pour ces deux mêmes périodes, passée et future.

8. Un ménage collectif comprend les communautés religieuses, les maisons de repos, les orphelinats, les logements pour étudiants ou travailleurs, les institutions hospitalières et les prisons. [25]

Parmi les variables qui influencent le plus le choix de migrer, on obtient que les couples mariés avec enfant(s) qui deviennent l'année suivante non mariés avec ou sans enfant(s), ou un ménage monoparental ou isolé, ont 20 à 30 fois plus de chance de déménager que ceux dont la structure du ménage n'a pas changé. Par exemple, passer du statut d'enfant au statut de chef de ménage ou conjoint correspond à une chance de migrer qui se situe entre 11 et 16 fois plus important que sans changement. De plus, les locataires dans le secteur privé ont 4 fois plus de chance de migrer que les propriétaires. Concernant la classe d'âge, les individus de 75 ans et plus sont ceux qui déménagent le moins alors que les plus mobiles sont les jeunes de moins de 18 ans.

3.4.2 Modèle de localisation

Maintenant que nous pouvons prédire si l'individu va migrer ou non, nous pouvons appliquer le modèle de localisation dans le cas où l'individu choisit de changer de lieu de résidence. Ce modèle utilise la sortie du modèle de propension à migrer afin de savoir sur quel individu il doit être appliqué. Il assigne alors une nouvelle commune de résidence à chaque individu qui va déménager. Celle-ci sera utilisée pour établir une nouvelle demande en mobilité.

3.4.2.1 Méthodologie

La méthode utilisée pour modéliser la localisation est le choix discret [27] car celle-ci offre plus de possibilités quand il y a beaucoup d'alternatives. Chaque fois qu'un individu quitte sa commune et déménage, il a 588 alternatives possibles comme choix de nouvelle commune de résidence. Cette technique consiste à déterminer l'utilité de chaque alternative et ensuite à calculer la probabilité de choisir chaque alternative. On suppose que l'individu va choisir l'alternative qui possède la plus grande utilité. Dans notre cas, les alternatives correspondent aux 589 communes belges, excepté la commune de résidence actuelle de l'individu puisqu'on regarde les migrations entre communes différentes.

Pour chaque alternative, l'utilité est une combinaison de termes constants et de variables explicatives pondérées par des paramètres qui seront estimés grâce au choix discret (modèle logit) via le logiciel BIOGEME [5]. Les variables explicatives peuvent être liées aux individus ou aux alternatives, c'est-à-dire aux communes. L'utilité se formule donc de la manière suivante :

$$U_{in} = C_i + \sum_j PInd_{ij} * VarInd_{jn} + \sum_k PMun_k * VarMun_k + \epsilon_{in}$$

où :

- U_{in} : utilité de l'alternative i pour l'individu n
- C_i : terme constant spécifique à l'alternative
- $PInd_{ij}$: paramètres lié à la variable individuelle j et l'alternative i
- $VarInd_{jn}$: variables liées aux individus
- $PMun_k$: paramètres liés aux caractéristiques des communes
- $VarMun_k$: variables liées aux communes

- ϵ_{in} : terme aléatoire inobservable

Les données utilisées pour calibrer le modèle sont les mêmes que celles du modèle de propension à migrer avec, en plus, des informations sur les accessibilités des communes, fournies par le modèle d'accessibilité.

3.4.2.2 Modèle

Le modèle logit de choix discret va donner une probabilité de choisir chaque alternative pour chaque individu. Les variables explicatives du modèle de localisation peuvent se classer en deux catégories :

- variables liées aux individus ou aux ménages : âge, type du ménage, niveau d'éducation et nationalité.
- variables liées aux alternatives ou communes : indicateurs d'accessibilité par rapport à l'emploi et aux services, indicateurs de conditions de vie par rapport à l'environnement, au logement et aux aspects socio-économiques, la distance entre la nouvelle commune de résidence et l'ancienne, la population de la nouvelle commune de résidence et un indicateur de l'immobilier.

La variable la plus significative est la distance par rapport à l'ancienne commune de résidence. En effet, on observe un nombre plus important de migrations sur des plus courtes distances. Pour plus de facilité, on a décidé d'utiliser pour chaque variable correspondante, un paramètre générique pour toutes les alternatives. De cette manière, les paramètres pour chaque variable influencent de la même façon chaque alternative et le nombre de paramètres est considérablement diminué.

On définit des variables de contextualisation pour les caractéristiques individuelles afin d'éviter d'obtenir les mêmes valeurs pour les variables explicatives qualitatives. En effet, on introduit dans la formule de l'utilité, des termes pour chaque catégorie correspondant au produit d'une variable binaire, indiquant si un individu appartient ou non à cette catégorie, et de la proportion représentée par cette catégorie dans la population de la commune. Prenons comme exemple, les classes d'âges des individus. La partie de l'utilité concernant cette variable sera constituée de 7 termes, un pour chaque classe d'âges. Par exemple, pour la classe des moins de 18 ans, ce terme sera de la forme suivante :

$$P_Age_0_18 * (Age_0_18 * AgeMun_0_18)$$

où :

- $P_Age_0_18$: paramètre lié à la variable Age_0_18
- Age_0_18 : variable binaire qui vaut 1 si l'individu a moins de 18 ans, et 0 sinon
- $AgeMun_0_18$: pourcentage d'individus ayant moins de 18 ans dans la population de la

commune considérée (dépend de l'alternative)

Ces variables sont significatives et indiquent que les individus tendent à s'installer dans des communes où la population possède les mêmes caractéristiques individuelles. Cela témoigne de l'envie des individus de vivre et s'installer près d'individus qui ont des caractéristiques socio-économiques semblables.

3.5 Modèles d'évolution

L'objectif final de MOBLOC étant d'utiliser les modèles décrits ci-dessus afin de pouvoir faire des prévisions, il est nécessaire de faire évoluer toutes les entrées des modèles. L'évolution temporelle sera simulée au travers de processus d'évolution temporelle implémentés ou qu'il faut encore implémenter dans VirtualBelgium. Nous avons expliqué ces processus dans le chapitre précédent à la section 2.2.

Chapitre 4

Migration résidentielle

Dans ce chapitre, nous allons expliquer en détails la manière dont nous avons implémenté le modèle de migration résidentielle issu de MOBLOC. Nous expliquerons également les problèmes rencontrés face aux différences concernant la définition des attributs des individus dans le programme VirtualBelgium par rapport à ceux issus de MOBLOC.

4.1 Modèle de propension à migrer

Le premier modèle qui constitue la migration résidentielle est celui de propension à migrer. La méthode utilisée pour ce modèle est la régression logistique binaire, comme nous l'avons détaillé dans le chapitre précédent. Nous allons expliquer le concept de régression logistique puis, nous expliquerons comment nous avons implémenté chaque variable explicative du modèle ainsi que les coefficients de régression correspondants.

4.1.1 Régression logistique binaire¹

La régression est utilisée afin de démontrer la dépendance entre la variable à expliquer et une ou plusieurs variables explicatives. Elle est dite « logistique » car la loi de probabilité est modélisée par une loi logistique.

La régression logistique est une technique de modélisation qui vise à construire un modèle permettant de prédire ou expliquer les valeurs prises par une variable qualitative à partir d'un ensemble de variables explicatives quantitatives ou qualitatives. Dans la cas où cette variable est binaire, elle prend uniquement comme valeur 1 (pour la modalité positive) ou 0 (pour la modalité négative) et la régression est dite « binaire ». Si la variable possède plus de deux modalités, la régression est dite « polytomique ».

1. Cette section sur la régression logistique binaire provient des références [23, 29].

Notations

La variable aléatoire à prédire, appelée variable dépendante ou expliquée, est notée Y . Dans notre cas, elle représente le fait de migrer entre l'année actuelle et l'année suivante pour un individu ω . Comme celle-ci est binaire, elle possède seulement deux modalités : 0 (si l'individu ne déménage pas) ou 1 (si l'individu déménage). Notons $Y(\omega)$ la valeur prise par Y pour un individu ω .

Supposons qu'on ait N variables prédictives ou explicatives, notées $\{X_1, X_2, \dots, X_N\}$. Alors, $X(\omega) = (X_1(\omega), X_2(\omega), \dots, X_N(\omega))$ représente le vecteur des valeurs de ces variables pour un individu ω . Dans notre cas, toutes ces variables explicatives sont binaires et le vecteur $X(\omega)$ comporte uniquement des 0 et des 1. Les variables qui influencent significativement la variable dépendante seront détaillées à la section 4.1.2.

Hypothèse fondamentale

Pour appliquer cette méthode de prédiction, nous devons calculer les probabilités conditionnelles $P[Y(\omega) = y \mid X(\omega)]$ pour chaque modalité $y \in \{1, 0\}$ de Y . Celles-ci représentent les probabilités d'un individu ω de prendre la modalité y sachant les valeurs prises par les variables explicatives $X(\omega)$. Grâce à ces probabilités, nous pourrions introduire les hypothèses de la régression logistique binaire.

D'après le théorème de Bayes [29], on a la formule suivante :

$$P[Y = y \mid X] = \frac{P[Y = y] \times P[X \mid Y = y]}{P[Y = 0] \times P[X \mid Y = 0] + P[Y = 1] \times P[X \mid Y = 1]} \quad (4.1)$$

où les ω ont été omis afin que ce soit plus lisible. Grâce à l'équation (4.1), nous pouvons calculer le rapport entre les deux probabilités conditionnelles :

$$\frac{P[Y = 1 \mid X]}{P[Y = 0 \mid X]} = \frac{P[Y = 1]}{P[Y = 0]} \times \frac{P[X \mid Y = 1]}{P[X \mid Y = 0]} \quad (4.2)$$

À partir de l'équation (4.2), on peut établir une règle de décision qui est la suivante :

$$\text{Si } \frac{P[Y = 1 \mid X]}{P[Y = 0 \mid X]} > 1 \text{ alors } Y = 1$$

En effet, si le rapport est supérieur à 1, cela signifie que la probabilité que Y soit égale à 1 sachant les valeurs de X est plus grande que la probabilité que Y soit égale à 0 sachant les valeurs de X . Autrement dit, Y a plus de chance de prendre la valeur 1. Si au contraire le rapport est inférieur à 1, Y aura plus de chance de prendre la valeur 0.

La régression logistique binaire introduit l'hypothèse fondamentale suivante :

$$\ln \left[\frac{P[X \mid Y = 1]}{P[X \mid Y = 0]} \right] = b_0 + b_1 X_1 + \dots + b_N X_N \quad (4.3)$$

où b_0, b_1, \dots, b_N sont les paramètres à estimer à partir des données.

Modèle LOGIT

On peut décrire d'une autre manière la régression logistique binaire. Soit $\pi(\omega) = P[Y(\omega) = 1 \mid X(\omega)]$, on appelle transformation LOGIT de $\pi(\omega)$ pour un individu ω , l'expression suivante :

$$\ln \left[\frac{\pi(\omega)}{1 - \pi(\omega)} \right] = a_0 + a_1 X_1(\omega) + \dots + a_N X_N(\omega) \quad (4.4)$$

où a_0, a_1, \dots, a_N sont les paramètres à estimer à partir des données.

Le rapport $\frac{\pi}{1-\pi} = \frac{P[Y=1|X]}{P[Y=0|X]}$ exprime un rapport de chances. En effet, si ce dernier vaut 2, cela signifie que l'individu aura deux fois plus de chances de choisir la valeur 1 pour Y que de choisir la valeur 0.

Posons $C(X) = a_0 + a_1 X_1 + \dots + a_N X_N$, on peut définir la fonction logistique :

$$\pi = \frac{e^{C(X)}}{1 + e^{C(X)}} \quad (4.5)$$

Si on injecte l'équation (4.5) dans l'équation (4.4), on retrouve bien le LOGIT $C(X)$. Ce dernier est défini entre $-\infty$ et $+\infty$, tandis que π est une probabilité qui est donc comprise entre 0 et 1. La représentation de la fonction logistique π se trouve à la figure 4.1.

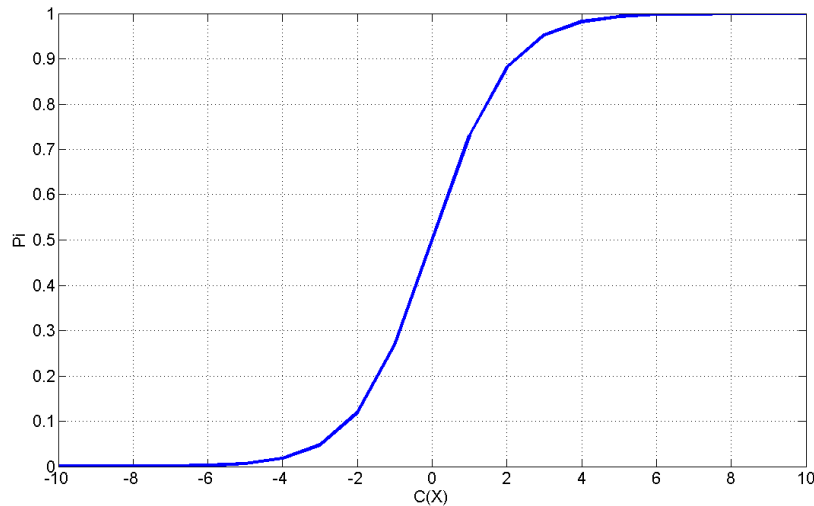


FIGURE 4.1 – Représentation de la fonction logistique [19]

$C(X)$ et π permettent de classer les individus selon leur propension à choisir la valeur 1 (positive). On peut établir la règle d'affectation suivante :

$$\text{Si } \frac{\pi}{1 - \pi} > 1 \text{ alors } Y = 1$$

$$\Leftrightarrow \text{Si } \pi > 0.5 \text{ alors } Y = 1$$

Grâce à la transformation LOGIT (4.4) et au fait que le logarithme népérien de valeurs supérieures à 1 soit toujours positif, la règle d'affectation devient la suivante :

$$\text{Si } C(X) > 0 \text{ alors } Y = 1$$

Sur la figure 4.1, on voit bien l'équivalence entre les différentes formulations de la règle d'affectation. En effet, lorsque $C(X)$ est positif, la fonction π est supérieure à 0.5. Dans ce cas, la probabilité étant plus grande que le seuil qui vaut 0.5, on obtient que $Y = 1$. Inversement, lorsque $C(X)$ est négatif, la probabilité π est inférieure au seuil de 0.5 et donc $Y = 0$.

Nous appliquerons cette dernière règle d'affectation lorsque nous appliquerons ce modèle sur notre population synthétique.

4.1.2 Variables explicatives

Comme ce modèle sert à déterminer si oui ou non l'individu va déménager entre l'année actuelle et l'année suivante, nous avons besoin de certaines caractéristiques concernant l'individu ainsi que certaines concernant le ménage auquel il appartient. Exposons à présent les différentes variables explicatives et les coefficients de régression correspondants, ainsi que la manière dont nous les avons implémentés au sein du programme VirtualBelgium.

4.1.2.1 Âge de l'individu

La première variable significative pour ce modèle est l'âge de l'individu. Ce dernier est défini sous forme de classes d'âges pour le modèle de propension à migrer issu de MOBLOC. Les individus sont également répartis sous différentes classes d'âges dans VirtualBelgium. La comparaison des valeurs de ces variables se trouve dans la table 4.1.

VARIABLES	VIRTUALBELGIUM	MOBLOC	COEFFICIENTS
Classe 1	0-5 ans	0-18 ans	1.3053
Classe 2	6-17 ans	19-29 ans	0.9055
Classe 3	18-39 ans	30-44 ans	0.5752
Classe 4	40-59 ans	45-54 ans	/
Classe 5	60 ans et plus	55-74 ans	-0.3473
Classe 6	/	75 ans et plus	-0.5436

TABLE 4.1 – Variables et coefficients pour les classes d’âges des individus

La classe d’âges de l’individu intervient dans la régression sous la forme d’une variable binaire qui vaut 1 si l’individu appartient à cette classe d’âges et 0 sinon. Afin d’initialiser la valeur de cette variable pour chaque individu de la population synthétique, nous allons regarder la variable dans VirtualBelgium qui correspond aux classes d’âges de l’individu. Cependant, comme nous pouvons le voir dans la table 4.1, les classes d’âges ne sont pas exactement définies de la même manière et il en existe une de plus dans MOBLOC. Nous ne pouvons donc malheureusement pas utiliser directement les classes d’âges définies dans VirtualBelgium pour définir les variables binaires dans la régression. Pour ce faire, nous allons nous baser sur la variable qui concerne l’âge et qui fait partie des attributs des individus de la population synthétique. Par exemple, pour la classe d’âges de 0 à 18 ans, la variable correspondante est initialisée de la manière suivante :

$$Age_0_18 = \begin{cases} 1 & \text{si l'âge de l'individu est compris entre 0 et 18} \\ 0 & \text{sinon} \end{cases}$$

Il existe une variable binaire pour chaque classe d’âges définie dans MOBLOC, excepté la classe 4. En effet, cette dernière a été retirée du modèle car celle-ci n’était pas significative pour la prédiction du choix de migrer ou non. Remarquons que parmi les 5 variables binaires, une seule d’entre elles prendra la valeur 1 en fonction de l’individu, puisque ce dernier ne peut évidemment appartenir qu’à une seule et unique classe d’âges. De cette manière, seul le coefficient correspondant à la classe d’âge de l’individu interviendra dans la régression.

Les coefficients de régression obtenus dans MOBLOC concernant les classes d’âges sont repris dans la table 4.1. On observe qu’il n’y a pas de coefficient correspondant à la classe 4 puisque, comme nous l’avons expliqué ci-dessus, la classe 4 de MOBLOC n’influait pas significativement le modèle et elle a donc dû être enlevée.

On remarque que le coefficient de régression est le plus élevé lorsque l’individu a moins

de 18 ans. Ce coefficient va ajouter un poids plus grand à la formule de régression de l'individu comparé aux individus qui appartiennent aux autres classes d'âges. Cela implique qu'un mineur aura une tendance à migrer plus forte comparée aux individus qui ont plus de 18 ans. Inversement, le plus petit coefficient correspond à la classe 6 qui forme la dernière classe d'âges. Autrement dit, les individus de plus de 75 ans auront plus de chance que les autres de ne pas déménager et de rester chez eux, ce qui semble tout à fait logique. De plus, on s'aperçoit que ce coefficient diminue avec l'âge, ce qui signifie que plus on vieillit, moins on aura tendance à déménager et on essayera sans doute de trouver un domicile à long terme.

Le code implémenté pour définir les classes d'âges dans VirtualBelgium se trouve à l'annexe C.1.1.

4.1.2.2 Genre de l'individu

La variable représentant le genre de l'individu influence également le choix de celui-ci concernant sa possible migration résidentielle. Les valeurs prises par cette variable dans MOBLOC et VirtualBelgium, ainsi que les coefficients de régression correspondants se trouvent à la table 4.2.

VARIABLES	VIRTUALBELGIUM	MOBLOC	COEFFICIENTS
Genre 1	Homme ('H')	Homme	-0.0593
Genre 2	Femme ('F')	Femme	/

TABLE 4.2 – Variables et coefficients pour le genre des individus

Tout d'abord, le genre de l'individu est défini par deux modalités qui sont bien entendu : « Homme » ou « Femme ». Cette définition est valable pour VirtualBelgium et MOBLOC. Cependant, on remarque que le modèle de régression n'a attribué qu'un seul coefficient pour le genre masculin et par conséquent, la variable du genre féminin n'intervient pas et ne possède pas de coefficient. D'après MOBLOC, les femmes ont une tendance à migrer plus faible que celle des hommes.

Comme la modalité « Femme » n'est pas intégrée à la régression, nous ne devons définir qu'une seule variable binaire de la manière suivante :

$$Genre = \begin{cases} 1 & \text{si le genre de l'individu est 'H' (Homme)} \\ 0 & \text{sinon} \end{cases}$$

Le coefficient correspondant à cette variable binaire est négatif mais il reste très proche

de 0, ce qui influencera très peu la valeur de la régression.

Le code implémenté pour définir la variable binaire qui représente le genre dans VirtualBelgium se trouve à l'annexe C.1.2.

4.1.2.3 Nationalité de l'individu

Concernant la variable qui représente la nationalité de l'individu, le programme VirtualBelgium ne prend pas encore en compte celle-ci et les individus sont donc tous considérés comme belges. Cependant, la modalité « Belge » pour la nationalité n'est pas considérée dans la régression d'après MOBLOC et elle n'est donc pas reliée à un coefficient de régression. Dans ce cas, et tant que la nationalité ne sera pas prise en compte pour la population synthétique, nous ne devrions pas définir de variable binaire

Les variables binaires qui correspondent aux autres nationalités ont été initialisées à 0 dans le programme et les coefficients liés à ces variables ont été encodés (annexe C.1.3). Si par la suite VirtualBelgium tient compte de la nationalité de l'individu, il suffira d'initialiser à 1 la variable binaire qui correspond.

4.1.2.4 Type et taille du ménage

Les variables concernant le type et la taille du ménage auquel appartient l'individu jouent un rôle significatif dans le choix de la migration résidentielle. Les types de ménage définis par VirtualBelgium et MOBLOC se trouvent dans la table 4.3. La taille du ménage est définie de la même manière pour les deux, à savoir le nombre d'individus qui appartiennent à ce ménage.

VARIABLES	VIRTUALBELGIUM	MOBLOC
Type 1	Couple ('C')	Couple (marié ou non) sans enfant
Type 2	Famille ('F')	Couple (marié ou non) avec enfant(s)
Type 3	Isolé ('I')	Isolé
Type 4	Monoparental ('N')	Monoparental
Type 5	/	Collectif
Type 6	/	Autre

TABLE 4.3 – Variables pour le type du ménage

Tout d'abord, nous remarquons que certains types de ménage ne sont pas représentés dans VirtualBelgium, tels que les ménages collectifs. De plus, MOBLOC tient compte du statut matrimonial en séparant les couples mariés et les couples non mariés, ce qui n'est

pas le cas de VirtualBelgium. En effet, le mariage n'est pas encore implémenté lors de l'évolution temporelle de la population synthétique. Afin de savoir comment définir les variables binaires nécessaires à la régression, nous avons besoin de connaître les variables synthétiques créées au sein de MOBLOC qui tiennent compte du type et de la taille du ménage. Ces variables sont définies dans la table 4.4 avec les coefficients de régression correspondants.

VARIABLES	TYPE	TAILLE	COEFFICIENTS
CSE2	Couple marié sans enfant	2	0.4458
CAE3	Couple marié avec enfant(s)	3	0.0574
CAE4		4	/
CAEp5		5	0.0634
CoS2	Couple non marié sans enfant	2	0.6759
CoA3	Couple non marié avec enfant(s)	3	0.6324
CoA4		4	0.5553
CoA5		5	0.6132
Iso1	Isolé	1	0.4052
Mon2	Monoparental	2	0.4489
Mon3		3	0.5567
Monp4		4	0.6016
Colm5	Collectif	5	-0.8215
Colp6		6	-0.3796
Autm3	Autre	3	-0.0534
Aut4		4	0.0500
Aut5		5	0.0399
Aut6		6	0.1119

TABLE 4.4 – Variables et coefficients pour le type et la taille du ménage issus de MOBLOC

Comme nous l'avons fait remarqué ci-dessus, il n'y a pas de ménages collectifs dans VirtualBelgium, ni de statut matrimonial. Cela entraîne que les variable binaires CSE2, CAE3, CAE4, CAEp5, Colm5 et Colp6 seront initialisées à 0 et ne vaudront jamais 1, puisque que pour cela, il faudrait que le ménage ait le type non marié ou collectif. Les variables restantes sont définies en fonction du type et de la taille du ménage issus de VirtualBelgium, comme détaillé à la table 4.5.

VARIABLES	TYPE	TAILLE	NOMBRE D'ENFANTS
CoS2	Couple ('C')	2	0
CoA3	Famille ('F')	3	1
CoA4		4	2
CoA5		5	3
Iso1	Isolé ('I')	1	0
Mon2	Monoparental ('N')	2	1
Mon3		3	1 ou 2
Monp4		4	1, 2 ou 3
Autm3	Famille ('F')	3	0
Aut4		4	1
Aut5		5	1 ou 2
Aut6		6	2, 3 ou 4

TABLE 4.5 – Définition des variables pour le type et la taille du ménage dans VirtualBelgium

Les variables définies dans la table 4.5, sont des variables binaires qui seront utilisées dans la régression. Elles valent 1 si l'individu appartient à un ménage qui possède le type et la taille correspondants et 0 sinon. Par exemple, pour un individu appartenant à un ménage de type 'N' et de taille 2, on a :

$$Mon2 = \begin{cases} 1 & \text{si le type du ménage de l'individu est 'N'} \\ & \text{et si la taille du ménage de l'individu est 2} \\ 0 & \text{sinon} \end{cases}$$

L'affectation des types de VirtualBelgium à ces variables a été évidente pour les variables CoS2 et Iso1, puisqu'il suffit de vérifier que le ménage est respectivement de type 'C' et 'I'. En effet, nous pouvons constater que lorsque le ménage est de type 'I', la taille de celui-ci est toujours de 1 puisque la personne est célibataire et lorsque son type est 'C', la taille est toujours de 2 puisqu'il s'agit d'un couple non marié sans enfant. L'affectation des variables Mon2, Mon3 et Monp4 a été réalisée également de manière évidente en vérifiant que le ménage est bien de type 'N' et que la taille vaut respectivement 2, 3 ou 4.

Concernant les variables restantes, nous avons décidé de manière intuitive de leur affecter le type 'F', puisqu'il s'agit de couples non mariés avec enfants ou d'autres configurations de ménage avec des enfants et/ou des adultes supplémentaires. Cependant, les variables CoA et Aut ont le même type 'F' et les mêmes tailles de ménage. Afin de les différencier, nous avons dû faire intervenir le nombre d'enfants dans le ménage qui fait

partie des attributs des ménages dans VirtualBelgium. Nous avons donc considéré que les variables CoA3, CoA4 et CoA5, représentant des couples non mariés avec enfant(s), décrivent des ménages qui possèdent respectivement 1, 2 et 3 enfants. Il n’y a donc aucun adulte supplémentaire dans ces ménages. En effet, nous trouvons plus pertinent que ce soient les variables Aut (autre) qui décrivent des familles avec enfant(s) et/ou adulte(s) supplémentaire(s). Néanmoins, pour ne pas confondre les variables CoA3 et Autm3, nous avons imposé comme condition, que la variable Autm3 décrive un ménage qui ne compte pas d’enfant. Or, ce cas n’existe pas dans la population d’origine de VirtualBelgium. En effet, si le ménage est de type ‘F’ ou de type ‘N’, on remarque qu’il y a toujours au moins un enfant et 0, 1 ou 2 adultes supplémentaires dans le ménage.

Notons que plusieurs types de ménage dans VirtualBelgium ne sont pas représentés dans le modèle de MOBLOC. Le type et la taille de ces ménages n’influenceront donc pas le choix de migrer des individus qui le composent. L’implémentation des variables binaires dans VirtualBelgium est détaillée à l’annexe C.1.4.

4.1.2.5 Évolution du type et de la taille du ménage

Les analyses de MOBLOC ont également permis de dire que l’évolution des variables synthétiques qui tiennent compte du type et de la taille du ménage auquel appartient l’individu, joue un rôle très important dans le choix de migrer ou non. Le nombre de variables représentant un changement de type et de taille dans le ménage entre l’année précédente et l’année actuelle est assez important. C’est pourquoi, nous allons uniquement donner les coefficients correspondants aux variables qui se trouvent dans la table 4.5, à savoir CoS, CoA, Iso, Mon et Aut. Ces coefficients sont décrits dans les tables 4.6 et 4.7.

VARIABLES	TYPE PRÉCÉDENT	TYPE ACTUEL	TAILLE	COEFFICIENTS
AutAutPlus	Aut	Aut	+	0.0756
AutAutMinus			–	0.3219
AutCoA	Aut	CoA	=	0.5391
AutCoAPlus			+	0.0853
AutCoAMinus			–	0.3702
AutCoS	Aut	CoS	/	0.2775
AutIso	Aut	Iso	/	0.4172
AutMon	Aut	Mon	=	0.4638
AutMonPlus			+	0.3144
AutMonMinus			–	0.3443

TABLE 4.6 – Variables et coefficients pour l’évolution du type et de la taille du ménage

VARIABLES	TYPE PRÉCÉDENT	TYPE ACTUEL	TAILLE	COEFFICIENTS
CoAAut	CoA	Aut	=	0.3727
CoAAutPlus			+	0.1251
CoAAutMinus			−	-0.2926
CoACoAPlus	CoA	CoA	+	0.1936
CoACoAMinus			−	0.1807
CoAIso	CoA	Iso	/	0.3373
CoAMon	CoA	Mon	=	0.0711
CoAMonPlus			+	0.3496
CoAMonMinus			−	0.2802
CoSAut	CoS	Aut	=	0.0151
CoSAutPlus			+	-0.2083
CoSCoAPlus	CoS	CoA	+	0.1710
CoSIso	CoS	Iso	/	0.0758
IsoAut	Iso	Aut	/	-0.0563
IsoCoA	Iso	CoA	/	-0.0482
IsoCoS	Iso	CoS	/	-0.0927
IsoMon	Iso	Mon	/	-0.0265
MonAut	Mon	Aut	=	0.1338
MonAutPlus			+	-0.1885
MonAutMinus			−	0.4181
MonCoA	Mon	CoA	=	0.2517
MonCoAPlus			+	-0.0149
MonCoAMinus			−	0.7542
MonCoS	Mon	CoS	/	0.1736
MonIso	Mon	Iso	/	0.4993
MonMonPlus	Mon	Mon	+	0.0991
MonMonMinus			−	0.2215

TABLE 4.7 – Variables et coefficients pour l'évolution du type et de la taille du ménage (2)

Afin de savoir si le type et la taille du ménage ont changé entre l'année précédente et l'année actuelle, nous avons besoin de garder en mémoire le type, la taille et le nombre d'enfants du ménage pour l'année précédente. Pour ce faire, nous avons ajouté trois attributs aux ménages : un de type « chaîne de caractères » pour le type du ménage et deux autres de type « entier » pour la taille et le nombre d'enfants du ménage. Lors de la première itération du processus d'évolution de la population synthétique, on n'a pas d'information sur leur valeur pour l'année précédente. On choisit donc de leur associer

la valeur actuelle du type, de la taille et du nombre d'enfants du ménage. Ensuite, après avoir appliqué le modèle de prédisposition à migrer, ces valeurs sont mises à jour afin de garder en mémoire les valeurs actuelles qui seront utilisées l'année d'après en tant que type, taille et nombre d'enfants de l'année précédente.

Ensuite, on peut définir une variable binaire nécessaire à la régression pour chaque changement de type et de taille du ménage, en tenant compte du type, de la taille et du nombre d'enfants précédents et actuels. Les coefficients de régression correspondants sont tous assez proches de 0 et n'influencent donc pas énormément la décision de migrer.

On peut faire exactement les mêmes démarches pour l'évolution entre l'année actuelle et l'année suivante du type et de la taille du ménage auquel appartient l'individu. Les coefficients de régression correspondants se trouvent dans les tables 4.8 et 4.9.

VARIABLES	TYPE PRÉCÉDENT	TYPE ACTUEL	TAILLE	COEFFICIENTS
AutAutPlus	Aut	Aut	+	1.2660
AutAutMinus			−	1.2808
AutCoA	Aut	CoA	=	1.9543
AutCoAPlus			+	2.5011
AutCoAMinus			−	1.5035
AutCoS	Aut	CoS	/	2.0040
AutIso	Aut	Iso	/	2.4476
AutMon	Aut	Mon	=	2.7014
AutMonPlus			+	1.9985
AutMonMinus			−	2.2865
CoAAut	CoA	Aut	=	1.2140
CoAAutPlus			+	0.2907
CoAAutMinus			−	1.3820
CoACoAPlus	CoA	CoA	+	0.3073
CoACoAMinus			−	1.0271
CoAIso	CoA	Iso	/	2.0133
CoAMon	CoA	Mon	=	1.9865
CoAMonPlus			+	2.8427
CoAMonMinus			−	1.5218

TABLE 4.8 – Variables et coefficients pour l'évolution du type et de la taille du ménage

VARIABLES	TYPE PRÉCÉDENT	TYPE ACTUEL	TAILLE	COEFFICIENTS
CoSAut	CoS	Aut	=	0.1904
CoSAutPlus			+	0.4686
CoSCoAPlus	CoS	CoA	+	0.6249
CoSIso	CoS	Iso	/	1.2800
IsoAut	Iso	Aut	/	1.5276
IsoCoA	Iso	CoA	/	1.6675
IsoCoS	Iso	CoS	/	1.4027
IsoMon	Iso	Mon	/	0.8687
MonAut	Mon	Aut	=	2.1226
MonAutPlus			+	1.9563
MonAutMinus			−	0.3060
MonCoA	Mon	CoA	=	1.9442
MonCoAPlus			+	1.2976
MonCoAMinus			−	2.1841
MonCoS	Mon	CoS	/	1.9334
MonIso	Mon	Iso	/	1.3664
MonMonPlus	Mon	Mon	+	0.8141
MonMonMinus			−	1.0001

TABLE 4.9 – Variables et coefficients pour l'évolution du type et de la taille du ménage (2)

Nous pouvons voir que les coefficients de régressions sont tous positifs, contrairement à ceux de l'évolution de l'année précédente à l'année actuelle. De plus, ils sont assez significatifs et influencent donc plus fortement le choix de migrer. Le coefficient le plus élevé correspond à la variable CoAMonPlus. Cela signifie que le ménage aura plus de chance de migrer si celui-ci était un couple avec enfant(s) l'année précédente et qui est devenu une famille monoparentale tout en augmentant la taille du ménage. On peut penser au cas où un des deux parents décide de quitter le ménage et un nouveau-né arrive dans le ménage ainsi qu'un membre supplémentaire qui pourrait être un grand-parent.

L'implémentation des variables représentant les évolutions du type et de la taille du ménage se trouvent à l'annexe C.1.5.

4.1.2.6 Position de l'individu au sein du ménage

Concernant la variable représentant la position de l'individu dans le ménage, celle-ci joue également un rôle important dans la migration résidentielle. Les différentes modalités

possibles pour cette variable sont définies pour VirtualBelgium et pour MOBLOC dans la table 4.10, ainsi que les coefficients de régression correspondants.

VARIABLES	VIRTUALBELGIUM	MOBLOC	COEFFICIENTS
Position 1	Chef de ménage ('H')	Chef de ménage	/
Position 2	Conjoint ('M')	Conjoint	-0.0326
Position 3	Enfant ('C')	Enfant	-0.3549
Position 4	Adulte supplémentaire ('A')	Autre	-0.8676

TABLE 4.10 – Variables et coefficients pour la position des individus dans le ménage

Pour VirtualBelgium, chaque modalité correspond à une variable de type « caractère », comme nous l'avons déjà expliqué dans le chapitre 2. Les modalités de la position dans le ménage sont pratiquement les mêmes dans MOBLOC, excepté le fait que la position 4 représente la modalité « Autre ». Néanmoins, cette modalité signifie qu'il s'agit de n'importe quelle position dans le ménage qui soit différente des 3 premières déjà définies. Nous pouvons donc supposer que celle-ci corresponde aux adultes supplémentaires, représentés par la position 'A' dans VirtualBelgium.

Les modalités étant équivalentes, nous pouvons définir les variables binaires utilisées dans la régression grâce aux valeurs des variables concernant la position qui sont définies dans VirtualBelgium. Cependant, nous remarquons dans la table 4.10, que la position 1 ne possède pas de coefficient de régression. En effet, la variable concernant le chef de ménage a été supprimée du modèle lors de la procédure effectuée dans MOBLOC car celle-ci ne l'améliorait pas. Nous devons donc définir 3 variables binaires pour la régression. Par exemple, la variable concernant la position 3 est initialisée de la manière suivante :

$$Position_enfant = \begin{cases} 1 & \text{si la position de l'individu dans le ménage est 'C'} \\ 0 & \text{sinon} \end{cases}$$

Les coefficients de régression concernant la position dans le ménage sont tous négatifs mais ils restent proches de 0. Le plus faible coefficient correspond à la position 4. Cela implique que l'individu de position 'A' comporte un poids négatif plus important par rapport aux individus d'autres positions. Ce terme va donc influencer négativement la valeur de la régression et l'individu aura donc une tendance légèrement plus faible à migrer. De la même manière, le terme correspondant au conjoint va influencer négativement la valeur de la régression, mais cette fois très légèrement puisque le coefficient correspondant est plus proche de 0. D'après ces coefficients, on peut estimer indépendamment des autres variables, que le conjoint aura plus de chance de migrer par rapport à l'enfant, qui lui

aura plus de chance de migrer par rapport à l'adulte supplémentaire dans le ménage. Nous supposons que cela est dû en grande partie à la croissance du nombre de divorces. Ensuite, les enfants quittent la maison familiale et enfin l'adulte supplémentaire peut rester plus longtemps dans le ménage, surtout s'il s'agit de personnes âgées qui vivent dans le ménage de leur enfant devenu adulte.

Le code qui initialise les variables et les coefficients de régression correspondants en fonction de la position de l'individu au sein du ménage, se trouve à l'annexe C.1.6.

4.1.2.7 Évolution de la position de l'individu

Tout comme le type et la taille du ménage, l'évolution de la position de l'individu au sein du ménage entre l'année précédente et l'année actuelle influence significativement le choix de l'individu quant à la migration résidentielle. Les variables qui représentent le changement de la position dans le ménage entre l'année précédente et l'année actuelle, ainsi que les coefficients de régression correspondants se trouvent dans la table 4.11.

VARIABLES	SIGNIFICATION	COEFFICIENTS
Position 12	Changement de 'H' à 'M'	-0.1419
Position 13	Changement de 'H' à 'C'	0.0449
Position 14	Changement de 'H' à 'A'	-0.3373
Position 21	Changement de 'M' à 'H'	-0.0449
Position 23	Changement de 'M' à 'C'	-0.1396
Position 24	Changement de 'M' à 'A'	-0.3872
Position 31	Changement de 'C' à 'H'	-0.3801
Position 32	Changement de 'C' à 'M'	-0.4479
Position 34	Changement de 'C' à 'A'	-0.0605
Position 41	Changement de 'A' à 'H'	-0.6197
Position 42	Changement de 'A' à 'M'	-0.2858
Position 43	Changement de 'A' à 'C'	-0.5364

TABLE 4.11 – Variables et coefficients pour l'évolution de la position des individus dans le ménage

Les différentes modalités pour la position de l'individu dans le ménage sont identiques à celles définies dans la section précédente. Afin de définir les variables binaires représentant ces changements de position, nous avons besoin de connaître la position de l'individu de l'année précédente. Pour ce faire, nous allons ajouter un attribut aux individus de la population synthétique dans VirtualBelgium qui conserve en mémoire la position au sein du ménage pour l'année précédente. Cet attribut est représenté par une variable de type

« caractère » qui est définie de la même manière que la position de l'individu dans le ménage pour l'année actuelle. Lors de la première itération du processus d'évolution de la population synthétique, on associe à cette variable la valeur de la position actuelle de l'individu dans le ménage, puisque nous n'avons aucune information à propos de celle-ci pour l'année précédente. Nous avons donc supposé pour la première année (2001), que la position de l'individu était la même l'année précédente (2000). Après avoir appliqué le modèle de propension à migrer, cette variable est mise à jour afin de garder en mémoire la position actuelle pour l'utiliser l'année suivante dans le modèle en tant que position précédente.

Ensuite, on définit une variable binaire nécessaire à la régression pour chaque changement de position dans le ménage décrit à la table 4.11, en tenant compte des positions précédente et actuelle. Par exemple, pour le changement de position 12 pour lequel l'individu est passé de chef de ménage à conjoint, la variable est définie de la manière suivante :

$$Position_12 = \begin{cases} 1 & \text{si la position précédente de l'individu dans le ménage est 'H'} \\ & \text{et si la position actuelle de l'individu dans le ménage est 'M'} \\ 0 & \text{sinon} \end{cases}$$

Les variables binaires sont définies de la même façon pour les autres changements de position. Concernant les coefficients de régression, ceux-ci influencent négativement le choix de migrer, excepté pour le changement de position 13, mais qui reste quand même fort proche de 0. Ce dernier représente le cas où un individu qui était chef de famille l'année précédente, devient un enfant l'année actuelle. Ce phénomène est assez rare mais il peut arriver lorsqu'un jeune individu qui a formé un nouveau ménage, pour une certaine raison, décide de revenir habiter chez ses parents. Dans ce cas, sa tendance à migrer à nouveau entre l'année actuelle et l'année suivante est plus élevée.

Le changement de position de l'individu au sein du ménage entre l'année actuelle et l'année suivante influence également significativement le modèle. Ces variables et les coefficients correspondants se trouvent dans la table 4.12. Elles sont définies de la même manière que pour l'évolution entre l'année précédente et l'année actuelle. La variable qui permet de déterminer la position de l'individu pour l'année suivante est définie après le processus d'évolution. De cette manière, on lui assigne la position actuelle de l'individu ou la nouvelle position si le processus prévoit une naissance entre cette année et l'année suivante.

VARIABLES	SIGNIFICATION	COEFFICIENTS
Position2 12	Changement de 'H' à 'M'	1.9532
Position2 13	Changement de 'H' à 'C'	2.1134
Position2 14	Changement de 'H' à 'A'	2.5333
Position2 21	Changement de 'M' à 'H'	1.0623
Position2 23	Changement de 'M' à 'C'	2.1068
Position2 24	Changement de 'M' à 'A'	2.2877
Position2 31	Changement de 'C' à 'H'	2.0952
Position2 32	Changement de 'C' à 'M'	2.6321
Position2 34	Changement de 'C' à 'A'	2.8335
Position2 41	Changement de 'A' à 'H'	2.6103
Position2 42	Changement de 'A' à 'M'	2.9585
Position2 43	Changement de 'A' à 'C'	2.6934

TABLE 4.12 – Variables et coefficients pour l'évolution de la position des individus dans le ménage

Nous pouvons remarquer que tous les coefficients de régression sont positifs avec des valeurs plus éloignées de 0 que pour l'évolution passée de la position. Cela signifie qu'un changement de position pour l'année suivante, suite à des événements prévus tels qu'une naissance ou un mariage, va influencer positivement le choix de migrer pour l'individu. Ces événements sont donc des facteurs très importants qui, une fois anticipés, permettent de prédire la migration de l'individu. L'implémentation des variables binaires représentant les évolutions de la position de l'individu dans le ménage se trouve à l'annexe C.1.7.

4.1.2.8 Migration précédente

Une variable qui influence également fortement le choix de migration résidentielle concerne la migration précédente, c'est-à-dire si l'individu a effectué une migration qui s'est déroulée entre l'année précédente et l'année actuelle. Les modalités de cette variable et les coefficients correspondants sont décrits dans la table 4.13 pour VirtualBelgium et MOBLOC.

VARIABLES	VIRTUALBELGIUM	MOBLOC	COEFFICIENTS
Migration 1	/ (ajout : True)	Oui	0.6269
Migration 2	/ (ajout : False)	Non	/

TABLE 4.13 – Variables et coefficients pour la migration précédente

Cette migration précédente a pour modalité « Oui » si l'individu a déménagé entre l'année précédente et l'année actuelle, et la modalité « Non » dans le cas contraire. Cependant, cette variable n'est pas définie dans VirtualBelgium et nous allons donc devoir l'implémenter. Pour ce faire, nous allons ajouter un attribut concernant la migration précédente pour chaque individu de la population synthétique, afin de retenir si celui-ci a déménagé l'année précédente. On définit donc une variable de type « booléen » qui prend la valeur `true` si l'individu a déménagé entre l'année précédente et l'année actuelle, et la valeur `false` sinon.

Comme nous n'avons aucune donnée sur la migration précédente lorsque nous générons la population synthétique, nous avons supposé qu'aucun individu n'avait effectué de migration l'année précédente. Autrement dit, lors de la première itération (2001), nous initialisons à `false` l'attribut correspondant à la migration précédente pour tous les individus de la population synthétique de VirtualBelgium. Ensuite, après avoir fait évoluer temporellement cette population, on assigne une nouvelle valeur à cette variable qui est fournie par le modèle de propension à migrer. En effet, ce modèle est représenté par une méthode (voir annexe à la table B.1) qui renvoie la valeur `true` si la régression indique que l'individu va déménager et la valeur `false` dans le cas contraire. Cette méthode sera plus détaillée à la section 4.1.3. Par conséquent, il suffit donc d'assigner la valeur de la méthode à la variable représentant la migration précédente. Dans ce cas, le fait d'avoir déménagé sera conservé en mémoire et pourra être réutilisé dans le calcul de la valeur de la régression l'année suivante.

À présent, la variable étant définie dans VirtualBelgium, nous pouvons construire les variables binaires concernant la migration précédente qui interviennent dans le calcul de la valeur de la régression. Comme le fait de ne pas avoir migré entre l'année précédente et l'année actuelle n'influence pas significativement le modèle, la variable correspondant à la migration 2 a été enlevée du modèle et elle n'a donc pas de coefficient de régression. La seule variable binaire intervenant dans la régression est définie de la manière suivante :

$$Migration_true = \begin{cases} 1 & \text{si la migration précédente de l'individu vaut } \text{true} \text{ (oui)} \\ 0 & \text{sinon} \end{cases}$$

Le coefficient de régression correspondant est positif. Autrement dit, si l'individu a déménagé l'année précédente, sa tendance à déménager l'année suivante sera plus importante.

L'implémentation de cette variable est décrite à l'annexe C.1.8, ainsi que l'initialisation du coefficient de régression correspondant.

4.1.2.9 Niveau d'éducation de l'individu

La variable qui représente le niveau d'éducation de l'individu correspond au diplôme le plus élevé obtenu par cet individu. Les modalités de cette variable sont définies pour VirtualBelgium et MOBLOC à la table 4.14, ainsi que les coefficients de régression qui leur sont associés.

VARIABLES	VIRTUALBELGIUM	MOBLOC	COEFFICIENTS
Éducation 1	Aucune ('O')	Aucune ou primaire	-0.124
Éducation 2	Primaire ('P')		
Éducation 3	/	Secondaire inférieure	-0.0360
Éducation 4	Secondaire supérieure ('S')	Secondaire supérieure	0.0214
Éducation 5	Supérieure ('U')	Supérieure	/
Éducation 6	/	Inconnue	-0.1026

TABLE 4.14 – Variables et coefficients pour le niveau d'éducation des individus

Dans VirtualBelgium, il existe 4 niveaux d'éducation que nous avons déjà définis dans le chapitre 2 à la section 2.1.1. Cependant, MOBLOC a défini 5 niveaux qui ne sont pas exactement les mêmes. En effet, le premier niveau d'éducation de MOBLOC regroupe en fait les deux premiers niveaux de VirtualBelgium. C'est pourquoi, nous avons défini la variable binaire correspondante de la manière suivante :

$$Education_aucune_prim = \begin{cases} 1 & \text{si le niveau d'éducation de l'individu est 'O' ou 'P'} \\ 0 & \text{sinon} \end{cases}$$

Dans ce cas, les individus de la population synthétique ayant un niveau d'éducation de type 'O' ou 'P' seront reliés au même coefficient de régression qui correspond au niveau « Aucune éducation ou primaire » dans MOBLOC. Ensuite, concernant le niveau d'éducation secondaire, seul le niveau secondaire supérieur est défini pour les individus de la population synthétique, tandis que dans MOBLOC, le niveau secondaire inférieur intervient en plus du niveau secondaire supérieur. Nous avons choisi d'initialiser la variable binaire correspondant au niveau secondaire supérieur de la manière suivante :

$$Education_sec_sup = \begin{cases} 1 & \text{si le niveau d'éducation de l'individu est 'S'} \\ 0 & \text{sinon} \end{cases}$$

Pour l'instant, comme il n'y a pas d'individus ayant un niveau d'éducation secondaire inférieure dans la population synthétique, la variable binaire correspondante est initialisée à 0. Elle pourra valoir 1 si on ajoute le niveau d'éducation 3 dans VirtualBelgium.

Concernant le niveau d'éducation 5, celui-ci est défini de la même manière des deux côtés mais cette variable a été enlevée du modèle. On ne peut donc pas lui attribuer une variable binaire ni lui faire correspondre de coefficient de régression. Enfin, le dernier niveau d'éducation défini dans MOBLOC survient lorsqu'on ne connaît pas le diplôme de l'individu. Néanmoins, cela n'arrive jamais dans VirtualBelgium, ce qui implique que la variable binaire correspondante sera toujours égale à 0 et le coefficient correspondant n'interviendra pas dans la régression.

A propos des valeurs des coefficients de régression, nous pouvons dire que ceux-ci influencent négativement la décision de migrer, excepté lorsque l'individu a obtenu comme plus haut diplôme celui du niveau secondaire supérieur. Il faut noter que cette influence positive est vraiment très faible et proche de 0. Le coefficient le plus faible correspond aux individus n'ayant aucune éducation ou uniquement une éducation primaire. On peut penser qu'ils ne migrent pas souvent parce qu'ils n'ont pas les moyens de se le permettre, contrairement aux individus ayant un diplôme du niveau secondaire supérieur.

L'implémentation de ces variables binaires et l'initialisation des coefficients de régression correspondants se trouvent à l'annexe C.1.9.

4.1.2.10 Type d'occupation du logement

La dernière variable concerne le type d'occupation du logement du ménage. Les modalités que peut prendre cette variable dans VirtualBelgium et dans MOBLOC, ainsi que les coefficients de régression correspondants se trouvent dans la table 4.15.

VARIABLES	VIRTUALBELGIUM	MOBLOC	COEFFICIENTS
Occupation 1	/ (ajout : 'O')	Propriétaire	/
Occupation 2	/ (ajout : 'L')	Locataire d'un logement privé	0.6615
Occupation 3	/	Locataire d'un logement social	-0.3448
Occupation 4	/	Occupant d'un logement gratuit	0.2254
Occupation 5	/	Type d'occupation inconnu	0.2369

TABLE 4.15 – Variables et coefficients pour le type d'occupation du logement

MOBLOC a défini 5 types d'occupation du logement pour les ménages. Cependant, cette variable ne fait pas partie des attributs des ménages de la population synthétique de VirtualBelgium. Nous avons donc décidé de rajouter un attribut aux ménages de cette population qui comporte uniquement deux modalités : propriétaire ou locataire. Afin de définir la proportion de ménages qui sont propriétaires ou locataires de leur logement,

nous nous sommes basés sur les chiffres données par le document [25]. Ce dernier stipule qu'il y avait en 2001 en Belgique, 2.709.868 logements occupés par des propriétaires et 1.188.255 logements occupés par des locataires. Cela représente environ 70% de ménages propriétaires de leur logement et 30% locataires. Remarquons que lorsqu'on dit qu'un ménage est propriétaire de son logement, cela implique que chacun des individus appartenant à ce ménage est également considéré comme propriétaire. Notons que le cas où un ménage possède plusieurs logements n'est pas pris en considération. En effet, nous allons initialiser aléatoirement 70% des ménages en tant que propriétaires et 30% en tant que locataires en prenant pour hypothèse que chaque ménage ne possède qu'un logement. Pour ce faire, on définit une variable de type « caractère » qui représente le type d'occupation du logement et on tire un nombre aléatoire entre 0 et 1. Si ce dernier est supérieur à 0.7, la variable se voit assigner le type 'L' correspondant au locataire, sinon on lui assigne le type 'O' correspondant au propriétaire. Cette affectation est effectuée uniquement lors de la génération de la population synthétique. Lorsqu'il y a une naissance, le nouveau-né se voit attribuer le même type d'occupation du logement que le ménage auquel il appartient. Il aurait été intéressant de faire évoluer cette variable au cours du temps.

Comme la variable représentant le type 'O' a été enlevée du modèle, nous devons définir une seule variable binaire qui correspond au type 'L' de la manière suivante :

$$Occupation_locataire = \begin{cases} 1 & \text{si le type d'occupation du ménage est 'L'} \\ 0 & \text{sinon} \end{cases}$$

Le coefficient de régression qui correspond au type 'L' est positif et assez élevé. Il va donc influencer positivement et de manière significative le choix de migrer. Cela implique qu'un individu appartenant à un ménage qui est locataire de son logement aura de plus fortes chances de migrer entre l'année actuelle et l'année suivante, comparé aux individus appartenant à des ménages propriétaires de leur logement. En effet, lorsqu'on est locataire, rien ne nous retient de déménager si on trouve un emploi ailleurs. Par contre, si on est propriétaire on va réfléchir plus longtemps avant de vraiment vouloir déménager et souvent, on va rester là où on s'est installé.

L'implémentation de cette variable et l'initialisation des coefficients de régression est détaillée à l'annexe C.1.10.

4.1.3 Implémentation de la régression

Maintenant que nous avons bien défini chaque variable binaire ainsi que les coefficients de la régression, nous pouvons calculer la valeur de cette régression pour chaque individu. Pour ce faire, nous allons calculer le terme $C(X) = a_0 + a_1 X_1 + \dots + a_N X_N$ qui intervient

dans le calcul de la fonction logistique π décrite par l'équation (4.5) de la section 4.1.1. Les variables aléatoires X_1, \dots, X_N correspondent aux variables binaires que nous avons défini dans la section précédente et les paramètres a_0, a_1, \dots, a_N correspondent aux coefficients de régression définis par MOBLOC. Notons que le coefficient a_0 est un terme constant qui intervient toujours dans la régression, même si toutes les valeurs des variables sont nulles. Celui-ci vaut -2.6899 et donc il influence négativement la régression.

Comme nous l'avons déjà dit, la fonction logistique π est la probabilité conditionnelle que $Y = 1$ sachant X , où Y est la variable binaire à expliquer qui représente le fait de migrer ou non entre l'année actuelle et l'année suivante. Cette variable vaut 1 si l'individu va déménager et 0 sinon. Le calcul de la régression, c'est-à-dire de $C(X)$, s'effectue de la manière suivante :

$$\begin{aligned} C(X) = & a_0 + \text{age} + \text{genre} + \text{nationalite} + \text{type_taille} + \text{evo_type_taille} \\ & + \text{evo2_type_taille} + \text{position} + \text{evo_position} + \text{evo2_position} \\ & + \text{migration} + \text{education} + \text{occupation} \end{aligned}$$

Chaque paramètre de cette formule correspond à une somme de produits entre les variables binaires pour chaque modalité et les coefficients de régression correspondants. Par exemple, le terme pour l'âge de l'individu est défini de la manière suivante :

$$\begin{aligned} \text{age} = & p_age_0_18 * \text{age_0_18} \\ & + p_age_19_29 * \text{age_19_29} \\ & + p_age_30_44 * \text{age_30_44} \\ & + p_age_55_74 * \text{age_55_74} \\ & + p_age_75\text{etplus} * \text{age_75etplus} \end{aligned}$$

D'après la règle de décision définie dans la section 4.1.1, si $C(X) > 0$ alors $Y = 1$ et l'individu aura une plus grande tendance à migrer. Dans le cas contraire, $Y = 0$ et l'individu aura plutôt tendance à ne pas migrer. La méthode créée dans VirtualBelgium avec laquelle on implémente le calcul de $C(X)$, sert à prédire la migration résidentielle d'un individu. Cette méthode renvoie une valeur de type « booléen » qui vaut **true** si $C(X) > 0$ et **false** sinon.

4.2 Modèle de localisation

Le second modèle de la migration résidentielle est le modèle de localisation. Si le modèle de propension à migrer prédit que l'individu va déménager, alors ce modèle fournit une nouvelle commune de résidence à l'individu. Comme nous l'avons déjà expliqué dans le chapitre 3, la méthode utilisée pour modéliser la localisation est le choix discret. Nous devons donc définir pour chaque individu, une utilité qui lui permettra de choisir une alternative parmi les 588 communes belges (excepté sa commune actuelle). La formulation de l'utilité peut être décomposée en 5 termes :

$$\text{Utilité} = \hat{\text{Âge}} + \text{Type_ménage} + \text{Nationalité} + \text{Education} + \text{Accessibilités}$$

Chacun de ces termes représente une somme de produits entre variables et paramètres et sont définis de la manière suivante pour l'alternative qui correspond à la commune de code INS égal à 11001 :

$$\begin{aligned} \hat{\text{Âge}} = & P_age0_18 * GA_0-18_11001 \\ & + P_age19_29 * GA_19-29_11001 \\ & + P_age30_44 * GA_30-44_11001 \\ & + P_age45_54 * GA_45-54_11001 \\ & + P_age55_64 * GA_55-64_11001 \\ & + P_age65_74 * GA_65-74_11001 \\ & + P_age75p * GA_75p_11001 \end{aligned}$$

où $GA_0-18_11001 = Age0-18 * \%Age0-18_11001$. La variable Age0-18 est binaire et vaut 1 lorsque l'individu a un âge inférieur à 18 ans, et 0 sinon. Le terme $\%Age0-18_11001$ représente le pourcentage d'individus ayant un âge inférieur à 18 ans et habitant dans la commune qui correspond au code INS 11001, à savoir Anvers.

$$\begin{aligned} \text{Type_ménage} = & P_men-iso * men-iso_11001 \\ & + P_men-mon * men-mono_11001 \\ & + P_men-cse * men-cse_11001 \\ & + P_men-cae * men-cae_11001 \\ & + P_men-aut * men-aut_11001 \end{aligned}$$

où le terme `men-iso_11001` désigne le pourcentage de ménages isolés dans la commune d'Anvers. Les autres termes sont définis de la même manière pour des ménages monoparentaux, des couples sans enfant, des couples avec enfant(s) et des ménages d'un autre type.

$$\begin{aligned}\text{Nationalité} = & P_nat_belge * nat_belge_11001 \\ & + P_nat_limitr * nat_limitr_11001 \\ & + P_nat_union * nat_union_11001 \\ & + P_nat_autre * nat_autre_11001\end{aligned}$$

où le terme `nat-belge_11001` désigne le pourcentage d'individus ayant la nationalité belge à Anvers. Les autres termes sont définis de la même manière pour les pays limitrophes, l'union européenne ou un autre pays.

$$\begin{aligned}\text{Education} = & P_instr_sec_inf * Instr_sec_inf_11001 \\ & + P_instr_sec_sup * Instr_sec_sup_11001 \\ & + P_instr_super * Instr_super_11001 \\ & + P_instr_autr * Instr_instr_autr_11001\end{aligned}$$

où le terme `Instr_sec-inf_11001` désigne le pourcentage d'individus d'Anvers qui ont comme diplôme le plus élevé, celui du niveau secondaire inférieur. Il en va de même pour le niveau secondaire supérieur, supérieur ou autre.

$$\begin{aligned}\text{Accessibilités} = & P_ICV_Log * ICV_Log_11001 + P_ICV_Env * ICV_Env_11001 \\ & + P_ICV_SE * ICV_SE_11001 + P_Immo * Immo_11001 \\ & + P_Pop * Pop_11001 + P_Dist * Distance_11001 \\ & + P_Acc_Empl_AgeAct * Acc_Empl_AgeAct_11001 \\ & + P_Acc_EmplAgeNonAct * Acc_Empl_AgeNonAct_11001 \\ & + P_Acc_Serv_AgeAct * Acc_Serv_AgeAct_11001 \\ & + P_Acc_Serv_AgeNonAct * Acc_Serv_AgeNonAct_11001\end{aligned}$$

où le terme `ICV_Log_11001` désigne l'indicateur de conditions de vie de logement pour la commune d'Anvers. Parmi les autres termes, on a des indicateurs d'environnement, socio-économiques, d'immobilier, de prix de propriété, la distance entre les communes ainsi que l'accessibilité à l'emploi et aux services.

Les valeurs des paramètres utilisés dans ces définitions se trouvent dans la table 4.16.

PARAMÈTRES	VALEUR
P_Acc_Empl_AgeAct	-0.364
P_Acc_Empl_AgeNonAct	0.0516
P_Acc_Serv_AgeAct	0.0133
P_Acc_Serv_AgeNonAct	-0.0441
P_Dist	-0.0665
P_ICV_Env	1.33
P_ICV_Log	-3.13
P_ICV_SE	5.45
P_Immo	0.625
P_Pop	0.723
P_age0_18	-7.48
P_age19_29	17.0
P_age30_44	3.04
P_age45_54	-4.19
P_age55_64	34.6
P_age65_74	30.0
P_age75p	17.8
P_instr-autr	1.86
P_instr-sec-inf	6.73
P_instr-sec-sup	6.52
P_instr-super	4.94
P_men-aut	1.41
P_men-cae	1.05
P_men-cse	6.39
P_men-iso	12.0
P_men-mon	6.16
P_nat_autre	11.2
P_nat_belge	2.65
P_nat_limitr	6.36
P_nat_union	7.19

TABLE 4.16 – Paramètres du modèle de localisation

Ce modèle n'a pas pu être implémenté dans sa totalité par manque de temps suite à divers soucis de programmation et d'installation de la toute dernière version de VirtualBelgium. Pour l'instant, les individus qui déménagent se voient donc attribuer une nouvelle commune de résidence aléatoirement.

4.3 Résultats

Nous avons appliqué le modèle de migration résidentielle expliqué ci-dessus à la population synthétique de VirtualBelgium pour toute la Belgique. Comme cette population contient un très grand nombre d'individus, nous avons dû lancer le programme sur un cluster qui utilise la programmation parallèle et réduit donc le temps d'exécution du programme. Les résultats ci-dessous concernent toute la Belgique sur une évolution de 5 ans, de 2001 à 2005. Le temps d'exécution afin d'obtenir ces résultats est de 6 minutes et 12 secondes. Les différentes représentations dans cette section ont été réalisées grâce à Matlab [19].

4.3.1 Migrations

Sur les 10.262.160 d'individus qui constituent la population synthétique de VirtualBelgium, notre modèle prédit que 299.849 individus vont déménager entre l'année 2001 et l'année 2002. Cela représente environ 2,9% de la population, ce qui n'est pas loin des 4% donnés par MOBLOC. Rappelons que ces migrations résidentielles ne concernent que les déménagements d'une commune vers une commune différente. Nous ne pouvons donc pas observer les migrations au sein d'une même commune. De plus, on regarde si l'individu va changer de lieu de résidence entre le 1^{er} janvier 2001 et le 1^{er} janvier 2002. C'est pourquoi, nous ne pouvons pas voir non plus les migrations multiples qui peuvent se produire au cours de cette année.

Regardons à présent l'évolution du nombre de migrations sur une période de 5 ans pour la population belge. Celle-ci est représentée sur la figure 4.2.

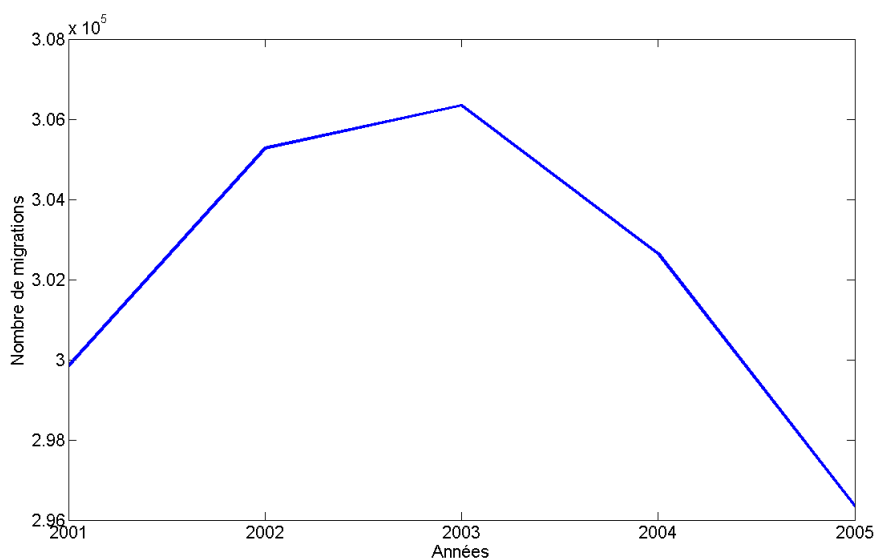


FIGURE 4.2 – Évolution du nombre de migrations en Belgique de 2001 à 2005

Nous remarquons que le nombre d'individus qui migrent augmente de 2001 à 2003 puis diminue assez fortement jusqu'en 2005. Le nombre de migrations sur ces 5 années est détaillé dans la table 4.18.

ANNÉES	NOMBRE DE MIGRATIONS
2001	299.849
2002	305.272
2003	306.343
2004	302.647
2005	296.340

TABLE 4.17 – Nombre de migrations en Belgique de 2001 à 2005

Malgré le fait que le nombre de migrations diminue en 2004 et en 2005, on observe que ces valeurs ne sont pas très éloignées l'une de l'autre. En effet, de 2003 à 2005 on constate qu'il y a environ 10.000 individus de moins parmi les migrants, ce qui représente seulement 3,33% de migrants en moins.

4.3.2 Migrations en fonction de la classe d'âges

Nous pouvons également observer le nombre d'individus qui ont migré entre 2001 et 2005, en fonction des classes d'âges auxquelles ils appartiennent. Ces nombres sont répertoriés dans la table 4.18.

ANNÉES	MIGRANTS 0-5 ANS	MIGRANTS 6-17 ANS	MIGRANTS 18-39 ANS	MIGRANTS 40-59 ANS	MIGRANTS 60 ANS ET PLUS
2001	24.348	57.205	105.784	65.003	47.509
2002	23.893	56.103	110.490	66.290	48.496
2003	22.300	54.606	111.035	67.744	50.658
2004	19.838	52.982	108.899	68.569	52.359
2005	17.041	51.032	105.915	68.824	53.528

TABLE 4.18 – Nombre de migrations en Belgique de 2001 à 2005

Nous remarquons que l'évolution du nombre de migrants en fonction des classes d'âges reste plus ou moins constante. On observe également une légère diminution de ce nombre pour les années 2004 et 2005, excepté pour les individus qui ont plus de 40 ans. Effectivement, le nombre de migrants ayant plus de 40 ans augmente au cours des années. Nous pouvons supposer que cela est dû en partie aux divorces qui touchent de plus en plus de gens de la population. Concernant les migrants de plus de 60 ans, nous supposons que

leur nombre augmente en raison du vieillissement de la population et du fait que ceux-ci doivent se rendre de plus en plus dans des maisons de repos, ce qui induit un changement de résidence. Afin de mieux visualiser ces nombres, nous avons représenté l'évolution de ceux-ci pour chaque classe d'âges à la figure 4.3.

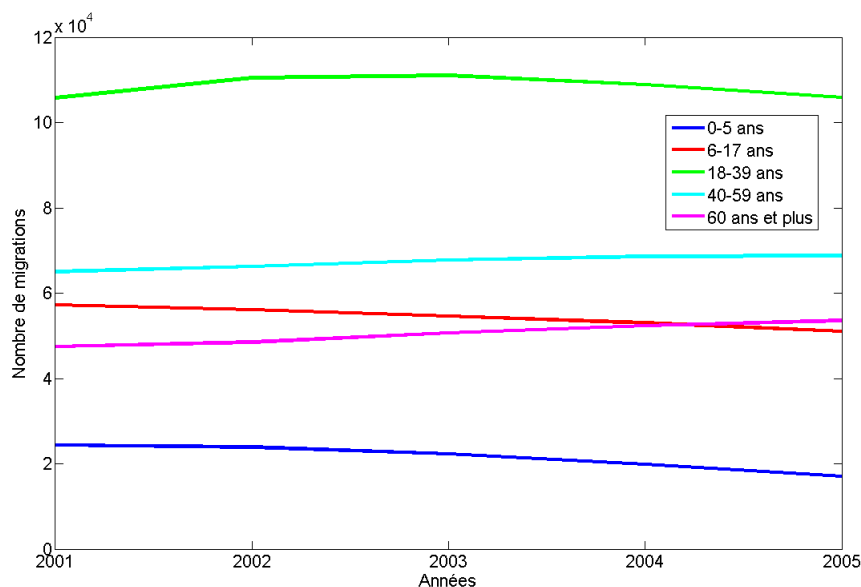


FIGURE 4.3 – Évolution du nombre de migrations en Belgique de 2001 à 2005 en fonction de l'âge

Parmi les individus qui ont déménagé entre 2001 et 2005, nous observons que la plus grande partie d'entre eux correspond aux individus dont l'âge est compris entre 18 et 39 ans. Tandis que les individus de moins de 5 ans, constituent la plus petite partie d'entre eux. Les autres individus qui ont déménagé sont répartis plus ou moins de la même manière dans les 3 classes d'âges restantes. Néanmoins, il faut tenir compte du fait que les classes d'âges ne regroupent pas le même nombre d'années. En effet, la classe d'âge qui regroupe les individus de moins de 5 ans ne pourra jamais contenir autant d'individus que la classe des 18-39 ans.

4.3.3 Migrations en fonction du genre

Nous pouvons ensuite nous intéresser au nombre de migrants en fonction du genre de l'individu. L'évolution de ce nombre est représentée à la figure 4.4.

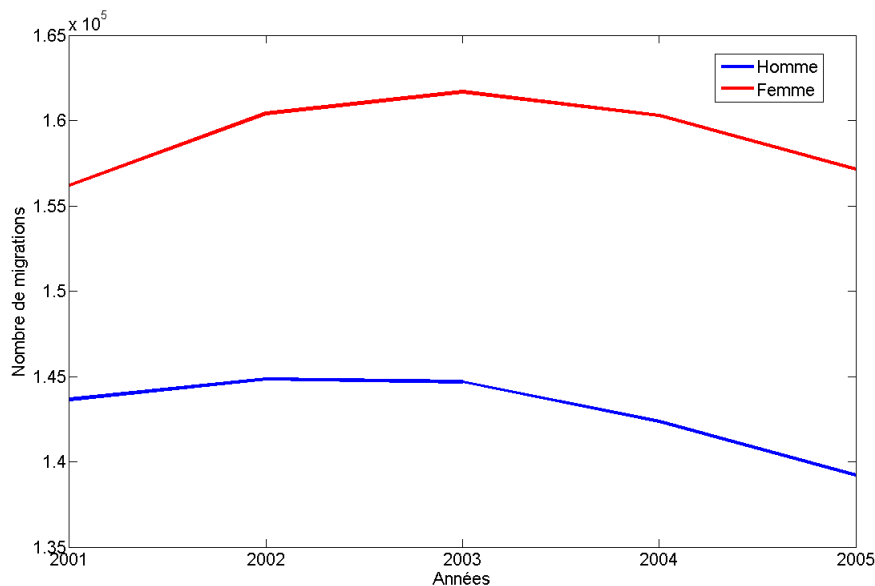


FIGURE 4.4 – Évolution du nombre de migrations en Belgique de 2001 à 2005 en fonction du genre

Nous pouvons remarquer qu'il y a une part plus importante de femmes parmi les individus qui ont migré entre 2001 et 2005. L'évolution du nombre de migrants évolue de la même manière pour les deux genres. Cependant, nous ne savons pas vraiment expliquer pourquoi le nombre de femmes migrantes est plus élevé que celui des hommes migrants. Cela peut être simplement dû au fait que la population à la base comptait plus de femmes que d'hommes.

4.3.4 Migrations en fonction du niveau d'éducation

Le nombre d'individus ayant migré entre 2001 et 2005 est représenté en fonction du niveau d'éducation de l'individu à la figure 4.5.

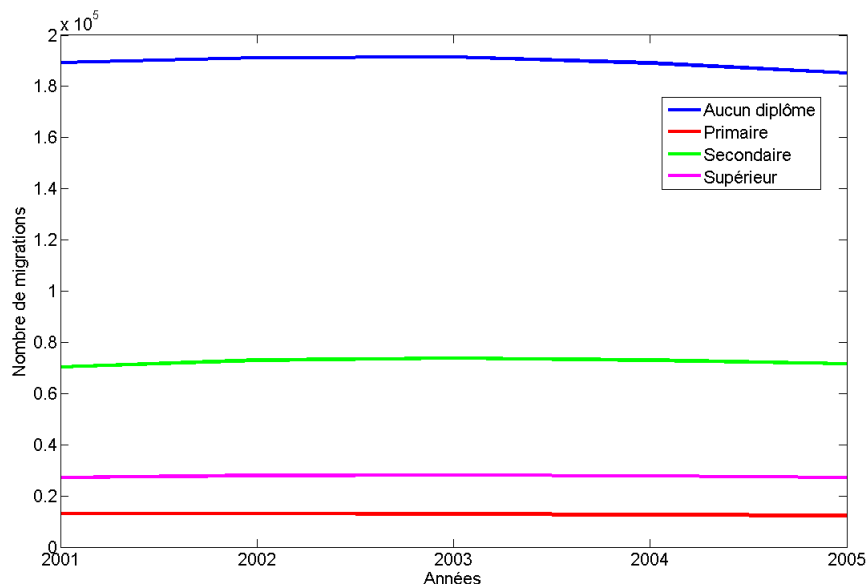


FIGURE 4.5 – Évolution du nombre de migrations en Belgique de 2001 à 2005 en fonction du niveau d'éducation

L'évolution du nombre de migrants reste assez stable au cours du temps pour chaque niveau d'éducation. De plus, nous pouvons constater que la plus grande partie de ces migrants concerne les individus qui n'ont pas de diplôme, suivie par ceux ayant un diplôme du niveau secondaire. Les individus qui ont un diplôme du niveau primaire constituent la plus petite partie. Néanmoins, cela pourrait sans doute s'expliquer par le fait qu'il y a très peu d'individus dans la population qui n'ont qu'un diplôme du niveau primaire. Inversement, une plus grande partie de ces migrants correspond aux individus sans diplôme. Nous pouvons supposer qu'il s'agit de personnes âgées puisqu'à leur époque, il n'était pas nécessaire d'avoir un diplôme très élevé. De plus, le nombre de personnes âgées devient de plus en plus important au sein de la population, ce qui pourrait expliquer le fait qu'il y ait autant de personnes sans diplômes parmi les migrants.

4.3.5 Migrations en fonction de l'obtention du permis de conduire

La représentation de l'évolution du nombre de migrants entre 2001 et 2005 est expliquée en fonction de l'obtention du permis de conduire à la figure 4.6.

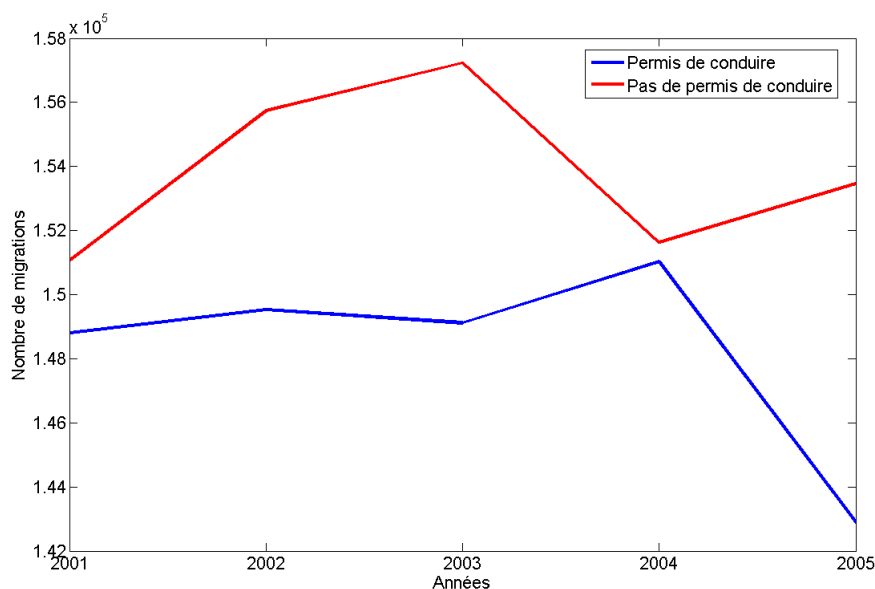


FIGURE 4.6 – Évolution du nombre de migrations en Belgique de 2001 à 2005 en fonction de l'obtention du permis de conduire

Tout d'abord, nous constatons que le nombre de migrants évolue de manière instable au cours du temps en fonction de l'obtention du permis de conduire. La majorité de ces migrants sont des individus qui n'ont pas de permis de conduire. Cela peut être dû au fait que ces personnes, ne pouvant pas se rendre à leur travail en voiture, sont obligées de trouver un lieu de résidence assez proche de leur travail et du centre-ville pour faire leur shopping. Dans ce cas, dès que l'individu changera de travail il aura tendance à déménager pour être à proximité de son nouveau lieu de travail.

Notons qu'on aperçoit une légère différence de comportement en 2004. En effet, cette année là, le nombre de migrants ayant un permis de conduire est pratiquement égal au nombre de migrants qui n'en ont pas.

4.3.6 Migrations en fonction de la position dans le ménage

Nous pouvons également observer l'évolution du nombre de migrants en fonction de la position de l'individu dans le ménage. Celle-ci est représentée à la figure 4.7.

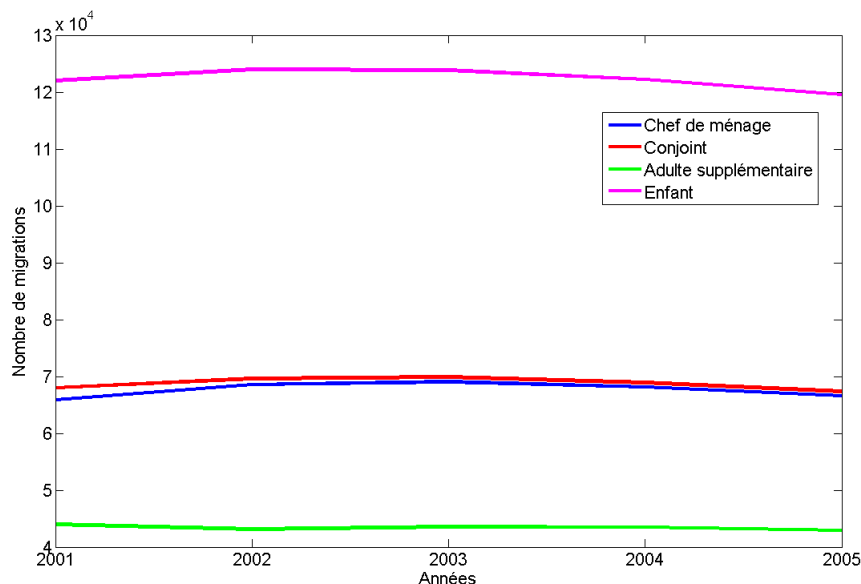


FIGURE 4.7 – Évolution du nombre de migrations en Belgique de 2001 à 2005 en fonction de la position dans le ménage

Premièrement, on remarque très clairement que les individus qui constituent la plus grande partie des migrants sont des enfants. En effet, une fois les études terminées, la suite logique pour un enfant consiste à trouver un emploi et à déménager afin de construire son propre ménage. La part de chefs de ménage et de conjoints parmi les migrants sont fort semblables. Cela pourrait s'expliquer par le fait que lors d'un divorce, un des deux membres du couple, ou parfois les deux membres déménagent. Dans ce cas, le nombre de chefs de ménages migrants augmente plus ou moins de la même manière que celui de conjoints migrants. Enfin, il y a un très faible part d'adultes supplémentaires parmi les individus migrants. Ce faible nombre peut provenir du fait qu'il est moins fréquent de trouver des individus de ce type au sein des ménages.

Conclusion

Ce mémoire a contribué à l'amélioration de la plateforme VirtualBelgium concernant la partie sur l'évolution temporelle. En effet, le processus de migration résidentielle n'était pas incorporé dans ce programme et les individus et les ménages de la population synthétique ne pouvaient donc pas changer de lieu de résidence. Les individus et les ménages de la population synthétique étaient reliés à un noeud du réseau représentant leur lieu de résidence qui restait inchangé au cours du temps.

Pour remédier à cet inconvénient, nous avons construit un modèle de migration résidentielle au niveau individuel. Ce dernier est issu du projet MOBLOC (Mobilities and Long Term Location Choices in Belgium) qui a étudié le choix de résidence fait par les ménages, la demande en transport et l'évolution de l'accessibilité qui en résulte. Après avoir lu attentivement les rapports de MOBLOC et compris à quoi servent chacun des modèles qui y sont décrits, nous avons tenté d'appliquer le modèle de migration résidentielle sur la population de VirtualBelgium. Ce modèle est divisé en deux autres modèles. Le premier prédit le choix résidentiel de l'individu, c'est-à-dire si l'individu va déménager entre l'année actuelle et l'année suivante. Le deuxième modèle, quant à lui, est un modèle de localisation qui sera appliqué sur l'individu qui va migrer, afin de lui fournir une nouvelle commune de résidence.

Le modèle de propension à migrer a été implémenté dans le programme VirtualBelgium. Pour ce faire, nous avons dû faire correspondre les variables définies dans MOBLOC avec les caractéristiques définissant les individus et les ménages de la population synthétique. Cependant, nous avons dû faire face à quelques difficultés lorsque les variables ne correspondaient pas. En effet, certaines variables importantes dans le choix de migration de l'individu n'étaient pas définies dans le programme, comme par exemple le type correspondant aux ménages collectifs. Celui-ci influence négativement le choix résidentiel mais aucun individu ne possède ce type dans VirtualBelgium. Il serait donc intéressant de définir de nouvelles modalités afin de « compléter » les variables caractérisant les individus et les ménages. D'autre part, l'évolution des attributs devrait être implémentée. En effet, un attribut tel que le plus haut diplôme obtenu ou l'obtention du permis de conduire, devrait évoluer au cours des années. Lorsqu'un enfant vient au monde, il n'a ni diplôme, ni

permis de conduire, mais sans ce modèle d'évolution, il ne les aura jamais. Cette évolution permettrait sans doute au modèle de choix résidentiel de fournir de meilleurs résultats. Néanmoins, nous avons obtenu comme résultat qu'environ 3% de la population belge déménageait entre 2001 et 2002, ce qui est assez proche des 4% définis dans MOBLOC.

Le modèle de localisation n'a pas été implémenté dans sa totalité par manque de temps et également par soucis de programmation et d'installation des logiciels nécessaires à l'utilisation de la dernière version de VirtualBelgium. Néanmoins, les variables et paramètres de ce modèle ont été décrits dans ce mémoire et la suite de l'implémentation de ce modèle ne devrait pas engendrer de difficultés. Pour l'instant, lorsqu'un individu déménage, on lui assigne aléatoirement une nouvelle commune de résidence. Il serait bien entendu plus réaliste de savoir la commune que le modèle de localisation pourrait proposer.

En conclusion, nous avons défini un modèle de propension à migrer qui nous donne des résultats satisfaisants. Cependant, il serait intéressant de comparer les résultats obtenus avec des données réelles sur le déménagement.

Bibliographie

- [1] Allison P., *Logistic Regression Using SAS : Theory and Application*, SAS Press, 1999
- [2] Bahri A., Carpentier S., Cornelis E., Eggerickx T., Gerber P., Klein S., Pauly X., Toint Ph., Walle F., *Mobilities and long term location choices in Belgium « MOBLOC »* - Final Report Phase 1, Brussels : Belgian Science Policy 2009 - 47 p. (Research Programme Science for a Sustainable Development), disponible sur http://www.belspo.be/belspo/ssd/science/Reports/MOBLOC%20-%20Final_Report%20-%20Phase1.pdf, consulté le 20/05/2014
- [3] Barthélemy J., Toint Ph., *Synthetic population generation without a sample*, Transportation Science 47 (2), pp 266-279, 2013
- [4] Barthélemy J., Toint Ph. (promoteur), Cornelis E. (promoteur), *A parallelized micro-simulation platform for population and mobility behaviour : application to Belgium*, Unamur : Faculté des sciences, Namur, 2014, Presses universitaires de Namur
- [5] Bierlaire, M. (2003). BIOGEME : A free package for the estimation of discrete choice models , *Proceedings of the 3rd Swiss Transportation Research Conference*, Ascona, Switzerland
- [6] Boigelot B., Programmation orientée-objet : notes de cours, Université de Liège, 2013
- [7] Carpentier S., Cornelis E., Dal L., Eggerickx T., Gerber P., Klein S., Pauly X., Toint Ph., Walle F., *Mobilities and long term location choices in Belgium « MOBLOC »* - Final Report, Brussels : Belgian Science Policy 2011 - 81 p. (Research Programme Science for a Sustainable Development), disponible sur http://www.belspo.be/belspo/ssd/science/Reports/MOBLOC_FinalReport%20ML.pdf, consulté le 20/05/2014

- [8] Carpentier S., Gerber P., Klein S., Omrani H., Vandenbulcke G., *Validation d'un modèle d'accessibilité par recoupement de données multi-sources. Application aux communes de Belgique*, annexe 1 de [7], disponible sur http://www.belspo.be/belspo/ssd/science/Reports/MOBLLOC_FinalReport_Annexes%201&2.pdf, consulté le 20/05/2014
- [9] Cornélis E., *Quels déterminants pour la mobilité résidentielle? Quels liens avec l'accessibilité?*, disponible sur <http://www.unamur.be/en/sci/naxys/archive/microsim/microsimfolder/paper3/view>, consulté le 20/05/2014
- [10] Debrand T., Taffin C., *Les facteurs structurels et conjoncturels de la mobilité résidentielle depuis 20 ans*, Économie et statistique n° 381-382, 2005, pp. 125-146
- [11] Denant-Boemont L., Université de Rennes, *Chapitre 3. Demande de trafic et congestion*, http://perso.univ-rennes1.fr/laurent.denant-boemont/transport_master1_chap3.pdf, consulté le 12/08/2014
- [12] Dico Info, <http://dictionnaire.phpmyvisites.net>, consulté le 5/11/2013
- [13] Futura-sciences : High-tech, <http://www.futura-sciences.com/magazines/high-tech>, 1995, consulté le 17/04/2014
- [14] Guermoudi M. A. et Benamar A., *Conception et implémentation d'un système multi-agent pour le test de primalité de nombre premier*, http://bibfac.univ-tlemcen.dz/bibfs/opac_css/doc_num.php?explnum_id=57, consulté le 08/08/2014
- [15] Henley A., *Residential mobility, housing equity and the labour market*, The Economic Journal, n° 108, 1998, pp. 27-414
- [16] Hubert J-P, Toint Ph, *La mobilité quotidienne des Belges*, Presses universitaires de Namur, 2002
- [17] Lehu J-M, *L'encyclopédie du marketing*, Editions d'Organisation, 2004, 956 p., disponible sur <http://www.e-marketing.fr/Definitions-Glossaire-Marketing/Modele-gravitaire-de-Reilly-loi-de-Reilly--10100.htm>, consulté le 10/08/2014

- [18] MATH@ES (Mathématiques appliquées en section ES), *Algorithme de Dijkstra*, http://yallouz.arie.free.fr/terminale_cours/graphes/dijkstra.php, consulté le 10/08/2014
- [19] MATLAB, The Language of Technical Computing, <http://www.mathworks.nl/products/matlab/>, consulté le 10/08/2014
- [20] MATSim, Multi-Agent Transport Simulation, <http://www.matsim.org/>, consulté le 20/05/2014
- [21] NOTRE BELGIQUE.BE, La Belgique, ses communes et ses dépendances, mise à jour le 7/11/12, 2008-2012, <http://www.notrebelgique.be/fr/index.php?nv=1&PHPSESSID=b8b0b83f9415a649fb4cfe4858a35fad>, consulté le 25/05/2014
- [22] OpenStreetMap, La carte coopérative libre, <http://www.openstreetmap.org/>, consulté le 20/05/2014
- [23] Rakotomalala R., *Régression Logistique et Pratique de la régression logistique*, Université de Lyon, Département informatique et statistique, <http://eric.univ-lyon2.fr/~ricco/cours/>, consulté le 10/08/2014
- [24] Repast, Recursive Porus Agent Simulation Toolkit, <http://repast.sourceforge.net/>, consulté le 20/05/2014
- [25] Statistics Belgium, http://statbel.fgov.be/fr/binaries/mono_200102_fr%5B1%5D_tcm326-35799.pdf, consulté le 25/06/2014
- [26] Sustainable Development & Mobility Solutions, *Le partage modal, c'est quoi?*, <http://sd-ms.jimdo.com/la-mobilit%C3%A9/le-partage-modal-c-est-quoi/>, consulté le 10/08/2014
- [27] Train K. E., *Discrete Choice Methods with Simulation*, Cambridge University Press, 2003, 334 p.
- [28] Van der Haegen H., Van Hecke E., Juchtmans G., *Les régions urbaines belges en 1991*, Études Statistiques, n° 104, 1996

- [29] Wikipédia : l'encyclopédie libre, <http://fr.wikipedia.org/>, consulté le 08/08/2014

Annexe A

Installation de VirtualBelgium

A.1 Installation

La section suivante, écrite en collaboration avec J. Barthelemy, décrit le processus d'installation du programme VirtualBelgium sur un ordinateur personnel. Plus précisément, ce document explique comment utiliser et installer VirtualBelgium sur une architecture GNU/Linux 64 bits. Pour commencer, nous détaillerons les fichiers et bibliothèques à télécharger. Ensuite, nous décrirons les commandes d'installation à entrer et enfin celles pour lancer le programme en lui-même.

A.2 Téléchargements

Avant toute chose, il est évident que nous devons commencer par télécharger une distribution Linux. Toutes les commandes décrites dans les lignes suivantes ont été testées sur la distribution Linux Mint 15.

Les fichiers du code en lui-même sont hébergés à l'adresse suivante :

<http://sourceforge.net/projects/virtualbelgium/files/>

et sont organisés de la manière suivante :

./	dossier racine, contient les scripts d'exécution ainsi que le "Makefile"
./bin	fichiers d'exécution et de configuration
./data	données d'entrées
./doc	documentation
./include	fichiers d'en-tête
./licenses	licences de Repast HPC, tinyxml2 et VirtualBelgium
./logs	fichiers log des simulations
./outputs	sorties générées par les simulations
./scripts	scripts pour le traitement des sorties
./src	fichiers sources et le "Makefile"

Listons à présent les différents outils obligatoires à la compilation et/ou exécution du programme.

Compilateur C++

Le projet étant codé en C++, un compilateur de ce langage est nécessaire. Le compilateur le plus connu est g++. Il est disponible dans le gestionnaire de dépôts grâce à la commande

```
apt-get install g++
```

Outil Make

Cet outil permet de simplifier les commandes de compilation lorsque le programme comporte beaucoup de fichiers. Il s'occupe des dépendances et automatise certains processus.

```
apt-get install make
```

La commande Make exécute les règles définies par le fichier Makefile.

L'environnement MPI

Le Message Passing Interface (MPI) est un standard permettant à différents processus d'une même exécution du programme de communiquer ensemble. Un utilitaire bien connu est l'exécutable OpenMPI, disponible à nouveau dans le gestionnaire de dépôt par la commande

```
apt-get install libopenmpi-dev openmpi-common
```

La librairie Repast HPC

La librairie principale utilisée par notre programme est le framework Repast HPC 1.0.1. Elle permet de modéliser au mieux un système basé sur le c++ et utilisant un grand ensemble d'unités de calcul. À l'origine, elle a été créée par le Laboratoire National d'Argonne. Le code compilable est disponible à l'adresse suivante : <http://repast.sourceforge.net>. Cette librairie en nécessite d'autres avant d'être compilée :

- Curl ;
- Boost 1.48 (ou supérieure) : En plus de cette librairie générale, nous avons besoin des spécifiques suivantes : boost-mpi, boost-system, boost-serialization and boost-filesystem ;
- NetCDF 4.2.1 ;
- NetCDF C++ 4.2.

Les commandes de téléchargement sont simplement :

Pour Curl :

```
apt-get install libcurl-dev
```

Pour Boost :

```
apt-get install libboost-dev libboost-mpi-dev libboost-serialization-dev  
libboost-system-dev libboost-filesystem-dev
```

Pour NetCDF :

```
apt-get install libnetcdf-dev
```

A.3 Repast HPC

Compilation de Repast HPC

Lorsque tout est téléchargé et extrait de l'archive, la compilation s'effectue grâce aux trois commandes suivantes : (attention à utiliser les privilèges super-utilisateur).

1. `./configure`
2. `./make`
3. `./make install`

Il se peut que la commande « `make` » génère une erreur. Pour y remédier, il faut remplacer toutes les occurrences de

```
getItems(...)
```

par

```
this->getItems(...)
```

dans les documents

```
./src/repast_hpc/DirectedVertex.h et ./src/repast_hpc/UndirectedVertex.h
```

Si une erreur se présente à nouveau lors de l'exécution du « `make` », le Makefile doit être édité de telle façon à supprimer toutes les références vers les modèles Zombie et Rumor.

A.4 Compilation et exécution de VirtualBelgium

Ordinateur personnel

Pour compiler simplement le programme sur un ordinateur personnel, il suffit d'exécuter la commande

```
make
```

Après la compilation, il faut lancer l'exécutable. Un script a été créé spécialement, il s'agit de la commande :

```
./run.sh NP
```

où NP est le nombre de processeurs désirés.

Exécution de calculs intensifs

Pour des simulations mettant en scène un nombre important d'agents, il est recommandé d'effectuer les opérations sur un système de plusieurs ordinateurs, « cluster ». Suivant les appareils utilisés, il est possible que le fichier « Makefile » doive être modifié. Si le cluster utilisé est Lemaitre du Consortium des Equipements de Calcul Intensif(<http://www.cec-hpc.be/>), la compilation se fait simplement avec la commande

```
make ucl
```

Ensuite, l'exécution se lance avec le script run_lemaitre2.sh

```
sbatch run_lemaitre2.sh
```

Ce script permet aussi de personnaliser l'exécution en proposant à l'utilisateur les options suivantes :

- mail-user : une adresse e-mail pour les notifications ;
- time : le temps d'exécution demandé ;
- ntask : le nombre de processeurs voulu ;
- mem-per-cpu : la mémoire réquisitionnée par processeur.

Debuggage

Si le code est modifié pour un ajout de fonctionnalité ou pour une optimisation, il est préférable de pouvoir le déboguer si une erreur survient lors de la compilation ou l'exécution. À nouveau la commande « make » va nous être utile en lui ajoutant l'option « debug »

```
make debug
```

Cette commande fait appel au débogger GNU gdb dont la documentation peut être trouvée à <http://www.sourceware.org/gdb/documentation/>.

A.5 Configuration de VirtualBelgium

VirtualBelgium est un programme de simulation par agent contenant plusieurs modèles possibles. Le choix du modèle se fait dans le fichier

```
\bin\model.props
```

Ce fichier reprend aussi le choix, entre autres, du réseau, de la population synthétique ou encore de la partie à simuler¹.

A.6 Mise à jour du code

Les dernières versions du code de VirtualBelgium sont disponibles sur un répertoire Subversion (SVN) hébergé à l'Université de Namur. Lorsque le logiciel Subversion est installé sur notre machine, les données sont récupérables en trois étapes :

1. Demander un compte à `virtualbelgium@math.unamur.be` ;
2. Créer un tunnel SSH vers Gauss :

```
ssh -N -f -l 5555 :localhost :3690 user@gauss.math.fundp.ac.be
```

3. Se connecter à

```
svn co svn ://user@localhost :5555/virtualbelgium/
```

pour recevoir la dernière version du programme

Les différentes commandes de Subversion pour éditer un projet sont listées à <http://svnbook.red-bean.com/index.en.html> (disponible en français).

1. Chaines d'activités, naissances, morts ou âges

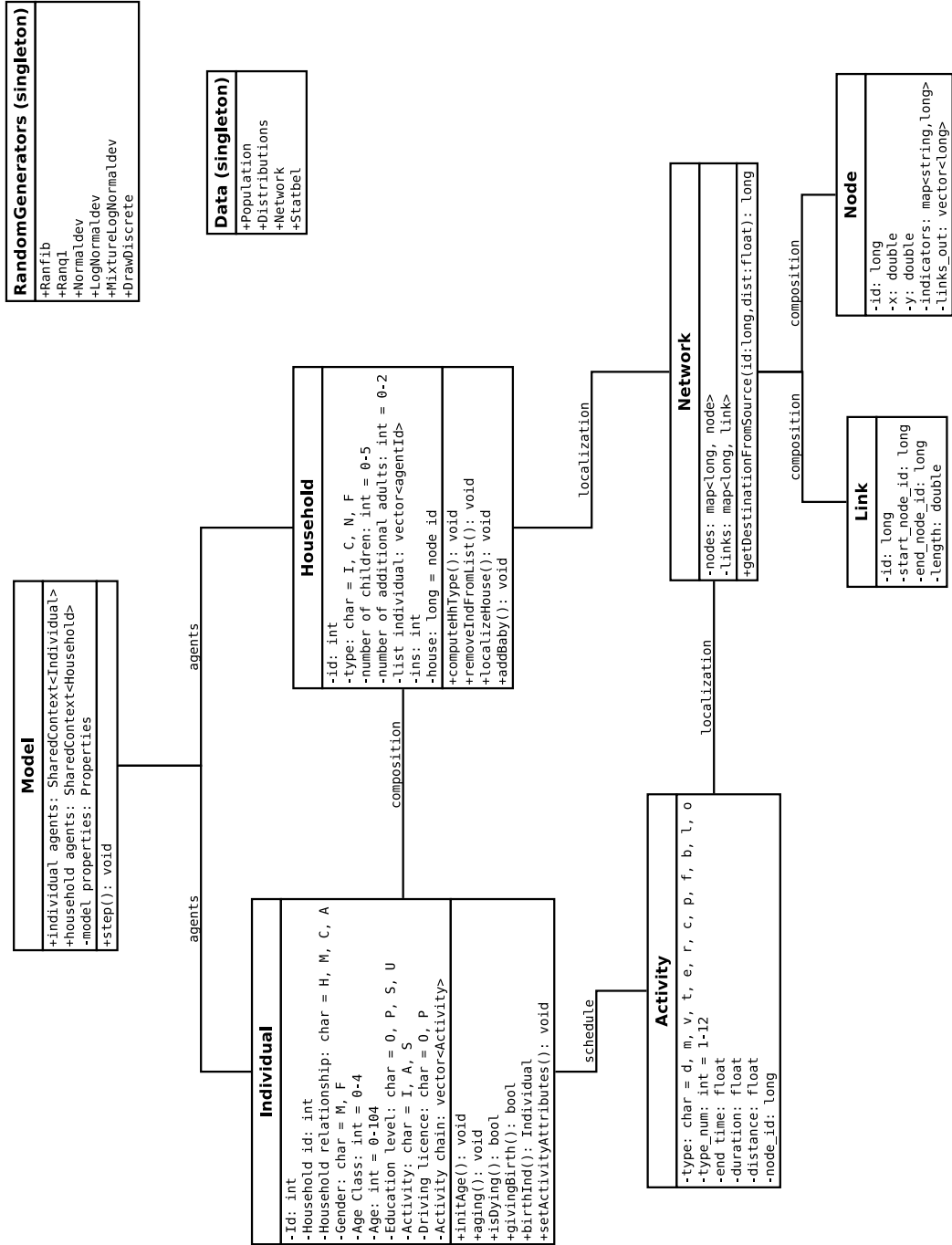


FIGURE A.1 – Diagramme des classes pour VirtualBelgium

Annexe B

Concepts de la programmation orientée objet

Cette annexe est principalement inspirée du cours programmation orientée-objet de Monsieur Bernard Boigelot[6], donné à l'université de Liège. La référence [29] a également été consultée.

La programmation orientée objet est une technique de programmation qui consiste à développer une partie d'un programme qui interagit avec les autres parties sans les connaître en détails. Plusieurs personnes peuvent donc travailler sur le même programme sans qu'elles connaissent en détails la partie de leurs collègues.

Cette technique de programmation possède son propre langage. Vous trouverez dans la table B.1, les définitions des différents concepts qui sont cités dans ce travail.

Concept	Définition
Classe	Ensemble d'attributs et de méthodes. Une classe est instanciée pour obtenir un objet. Elle représente la structure d'un objet.
Classe principale	Classe qui possède une méthode Main. Cette méthode permet d'exécuter le programme.
Instance d'une classe	Un objet.
Méthode	Opération réalisable par un objet. Elle peut être caractérisée de fonction d'une classe.
Objet	Structure de données présentes dans la mémoire de l'ordinateur au cours de l'exécution d'un programme.
Programme Orienté-objet	Ensemble de classes dont la classe principale.
Héritage	Création d'une classe qui sont dérivées d'autres classes et qui a donc accès aux variables et méthodes de ces dernières.
Single Singleton	Un concept en programmation qui permet une instance unique d'une classe.

TABLE B.1 – Définitions des concepts de la programmation orientée objet

Annexe C

Implémentation du modèle de migration résidentielle

Dans ce chapitre se trouvent les différents codes implémentés afin d'inclure au programme VirtualBelgium, un modèle de migration résidentielle.

C.1 Modèle de propension à migrer

C.1.1 Implémentation de l'âge

```
//initialisation des variables représentant les classes d'âges
int age_0_18 = 0;    //classe 1
int age_19_29 = 0;   //classe 2
int age_30_44 = 0;   //classe 3
int age_55_74 = 0;   //classe 5
int age_75more = 0;  //classe 6

//initialisation de la classe d'âges de l'individu
if (this->_age <= 18)
    age_0_18 = 1;
else if ((19 <= this->_age) && (this->_age <= 29))
    age_19_29 = 1;
else if ((30 <= this->_age) && (this->_age <= 44))
    age_30_44 = 1;
else if ((55 <= this->_age) && (this->_age <= 74))
    age_55_74 = 1;
else if (75 <= this->_age)
    age_75more = 1;
```



```
//coefficients de régression par classe d'âges
float p_age_0_18 = 1.3053;
float p_age_19_29 = 0.9055;
float p_age_30_44 = 0.5752;
float p_age_55_74 = -0.3473;
float p_age_75more = -0.5436;
```

C.1.2 Implémentation du genre

```
//initialisation des variables représentant le sexe
int _gender = 0; //si c'est une femme
if (this->_gender == 'H')
    _gender = 1; //si c'est un homme

//coefficient de régression pour sexe=homme
float p_gender = -0.0593;
```

C.1.3 Implémentation de la nationalité

```
//initialisation des variables représentant la nationalité
int nat_2 = 1;
int nat_3 = 0;
int nat_4 = 0;
int nat_5 = 0;
int nat_6 = 0;
int nat_7 = 0;
int nat_8 = 0;

//coefficients de régression par nationalité
float p_nat_2 = -0.5022;
float p_nat_3 = -0.0967;
float p_nat_4 = 0.1600;
float p_nat_5 = 0.0994;
float p_nat_6 = 0.2250;
float p_nat_7 = 0.1581;
float p_nat_8 = 0.1301;
```

C.1.4 Implémentation du type et de la taille du ménage

```
//initialisation des variables représentant le type et la taille du ménage
```

```
int hh_Aut4 = 0; //HhType = F
int hh_Aut5 = 0; //HhType = F
int hh_Aut6 = 0; //HhType = F
int hh_Autm3 = 0; //HhType = F
int hh_CAE3 = 0;
int hh_CAEp5 = 0;
int hh_CSE2 = 0;
int hh_CoA3 = 0; //HhType = F
int hh_CoA4 = 0; //HhType = F
int hh_CoA5 = 0; //HhType = F
int hh_CoS2 = 0; //HhType = C
int hh_Colm5 = 0;
int hh_Colp6 = 0;
int hh_Iso1 = 0; //HhType = I
int hh_Mon2 = 0; //HhType = N
int hh_Mon3 = 0; //HhType = N
int hh_Monp4 = 0; //HhType = N

//initialisation des variables en fonction du type et de la taille
if ((HhType == 'N') && (HhSize == 2))
    hh_Mon2 = 1;
else if ((HhType == 'N') && (HhSize == 3))
    hh_Mon3 = 1;
else if ((HhType == 'N') && (HhSize == 4))
    hh_Monp4 = 1;
else if (HhType == 'I')
    hh_Iso1 = 1;
else if (HhType == 'C')
    hh_CoS2 = 1;
else if ((HhType == 'F') && (HhSize == 3) && (NChildren == 1))
    hh_CoA3 = 1;
else if ((HhType == 'F') && (HhSize == 4) && (NChildren == 2))
    hh_CoA4 = 1;
else if ((HhType == 'F') && (HhSize == 5) && (NChildren == 3))
    hh_CoA5 = 1;
else if ((HhType == 'F') && (HhSize == 3))
    hh_Autm3 = 1;
else if ((HhType == 'F') && (HhSize == 4))
    hh_Aut4 = 1;
```

```
else if ((HhType == 'F') && (HhSize == 5))
    hh_Aut5 = 1;
else if ((HhType == 'F') && (HhSize == 6))
    hh_Aut6 = 1;

//coefficients de régression par type et taille
float p_hh_Aut4 = 0.0500;
float p_hh_Aut5 = 0.0399;
float p_hh_Aut6 = 0.1119;
float p_hh_Autm3 = -0.0534;
float p_hh_CAE3 = 0.0574;
float p_hh_CAEp5 = 0.0634;
float p_hh_CSE2 = 0.4458;
float p_hh_CoA3 = 0.6324;
float p_hh_CoA4 = 0.5553;
float p_hh_CoA5 = 0.6132;
float p_hh_CoS2 = 0.6759;
float p_hh_Colm5 = -0.8215;
float p_hh_Colp6 = -0.3796;
float p_hh_Iso1 = 0.4052;
float p_hh_Mon2 = 0.4489;
float p_hh_Mon3 = 0.5567;
float p_hh_Monp4 = 0.6016;
```

C.1.5 Implémentation de l'évolution du type et de la taille du ménage

```
//initialisation des nouvelles variables pour l'évolution du type et de
    la taille du ménage
float evo_hh_AutAutPlus = 0;
float evo_hh_AutAutMinus = 0;
float evo_hh_AutCAE = 0;
float evo_hh_AutCAEPlus = 0;
float evo_hh_AutCAEMinus = 0;
float evo_hh_AutCSE = 0;
float evo_hh_AutCoA = 0;
float evo_hh_AutCoAPlus = 0;
float evo_hh_AutCoAMinus = 0;
float evo_hh_AutCoS = 0;
```

```
float evo_hh_AutCol = 0;
float evo_hh_AutIso = 0;
float evo_hh_AutMon = 0;
float evo_hh_AutMonPlus = 0;
float evo_hh_AutMonMinus = 0;

float evo_hh_CAEEAut = 0;
float evo_hh_CAEEAutPlus = 0;
float evo_hh_CAEEAutMinus = 0;
float evo_hh_CAEECAEPlus = 0;
float evo_hh_CAEECAEMinus = 0;
float evo_hh_CAEECSE = 0;
float evo_hh_CAEECoA = 0;
float evo_hh_CAEECoAPlus = 0;
float evo_hh_CAEECoAMinus = 0;
float evo_hh_CAEECoS = 0;
float evo_hh_CAEECol = 0;
float evo_hh_CAEEIso = 0;
float evo_hh_CAEEMon = 0;
float evo_hh_CAEEMonPlus = 0;
float evo_hh_CAEEMonMinus = 0;

float evo_hh_CSEAlter = 0;
float evo_hh_CSEAlterPlus = 0;
float evo_hh_CSECAEPlus = 0;
float evo_hh_CSEIso = 0;

float evo_hh_CoAAAlter = 0;
float evo_hh_CoAAAlterPlus = 0;
float evo_hh_CoAAAlterMinus = 0;
float evo_hh_CoACAE = 0;
float evo_hh_CoACAEPlus = 0;
float evo_hh_CoACAEMinus = 0;
float evo_hh_CoACoAPlus = 0;
float evo_hh_CoACoAMinus = 0;
float evo_hh_CoAIso = 0;
float evo_hh_CoAMon = 0;
float evo_hh_CoAMonPlus = 0;
float evo_hh_CoAMonMinus = 0;
```

```
float evo_hh_CoSAlter = 0;
float evo_hh_CoSAlterPlus = 0;
float evo_hh_CoSCAEPlus = 0;
float evo_hh_CoSSE = 0;
float evo_hh_CoSCoAPlus = 0;
float evo_hh_CoSIso = 0;

float evo_hh_ColAlter = 0;
float evo_hh_ColCol = 0;

float evo_hh_IsoAut = 0;
float evo_hh_IsoCAE = 0;
float evo_hh_IsoCSE = 0;
float evo_hh_IsoCoA = 0;
float evo_hh_IsoCoS = 0;
float evo_hh_IsoCol = 0;
float evo_hh_IsoMon = 0;

float evo_hh_MonAlter = 0;
float evo_hh_MonAlterPlus = 0;
float evo_hh_MonAlterMinus = 0;
float evo_hh_MonCAE = 0;
float evo_hh_MonCAEPlus = 0;
float evo_hh_MonCAEMinus = 0;
float evo_hh_MonCoA = 0;
float evo_hh_MonCoAPlus = 0;
float evo_hh_MonCoAMinus = 0;
float evo_hh_MonCoupl = 0;
float evo_hh_MonIso = 0;
float evo_hh_MonMonPlus = 0;
float evo_hh_MonMonMinus = 0;

//initialisation de l'évolution du Type Aut l'année passée
if ((HhType_previous == 'F') && (((HhSize_previous == 3) &&
(NChildren_previous == 0)) || ((HhSize_previous == 4) &&
(NChildren_previous == 1)) || ((HhSize_previous == 5) &&
(NChildren_previous < 3)) || (HhSize_previous == 6))) {
    //AutAut
```

```
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 0))
    || ((HhSize == 4) && (NChildren == 1)) || ((HhSize == 5) &&
    (NChildren < 3)) || (HhSize == 6))) {
        if (HhSize > HhSize_previous)
evo_hh_AutAutPlus = 1;
        else {
if (HhSize < HhSize_previous)
    evo_hh_AutAutMinus = 1;
        }
    }
    //AutCoA
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 1))
    || ((HhSize == 4) && (NChildren == 2)) || ((HhSize == 5)
    && (NChildren == 2)))) {
        if (HhSize > HhSize_previous)
evo_hh_AutCoAPlus = 1;
        else {
if (HhSize < HhSize_previous)
    evo_hh_AutCoAMinus = 1;
else
    evo_hh_AutCoA = 1;
        }
    }
    //AutCoS
    if (HhType == 'C') {
        evo_hh_AutCoS = 1;
    }
    //AutIso
    if (HhType == 'I') {
        evo_hh_AutIso = 1;
    }
    //AutMon
    if (HhType == 'N') {
        if (HhSize > HhSize_previous)
evo_hh_AutMonPlus = 1;
        else {
if (HhSize < HhSize_previous)
    evo_hh_AutMonMinus = 1;
else
```

```

    evo_hh_AutMon = 1;
    }
    }
}

if ((HhType_previous == 'F') && (((HhSize_previous == 3) &&
    (NChildren_previous == 1)) || ((HhSize_previous == 4) &&
    (NChildren_previous == 2)) || ((HhSize_previous == 5) &&
    (NChildren_previous == 2)))){
    //CoAAut
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 0))
        || ((HhSize == 4) && (NChildren == 1)) || ((HhSize == 5) &&
            (NChildren < 3)) || (HhSize == 6)))){
        if (HhSize > HhSize_previous)
evo_hh_CoAAAlterPlus = 1;
        else {
if (HhSize < HhSize_previous)
    evo_hh_CoAAAlterMinus = 1;
else
    evo_hh_CoAAAlter = 1;
        }
    }
    //CoACoA
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 1))
        || ((HhSize == 4) && (NChildren == 2)) || ((HhSize == 5)
            && (NChildren == 2)))){
        if (HhSize > HhSize_previous)
evo_hh_CoACoAPlus = 1;
        else {
if (HhSize < HhSize_previous)
    evo_hh_CoACoAMinus = 1;
        }
    }
    //CoAIso
    if (HhType == 'I'){
        evo_hh_CoAIso = 1;
    }
    //CoAMon

```

```
    if (HhType == 'N'){
        if (HhSize > HhSize_previous)
evo_hh_CoAMonPlus = 1;
        else {
if (HhSize < HhSize_previous)
    evo_hh_CoAMonMinus = 1;
else
    evo_hh_CoAMon = 1;
        }
    }
}

if (HhType_previous == 'C'){
    //CoSAlter
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 0))
        || ((HhSize == 4) && (NChildren == 1)) || ((HhSize == 5)
        && (NChildren < 3)) || (HhSize == 6)))){
        if (HhSize > HhSize_previous)
evo_hh_CoSAlterPlus = 1;
        else {
if (HhSize == HhSize_previous)
    evo_hh_CoSAlter = 1;
        }
    }
    //CoSCoA
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 1))
        || ((HhSize == 4) && (NChildren == 2)) || ((HhSize == 5)
        && (NChildren == 2))))) {
        if (HhSize > HhSize_previous)
evo_hh_CoSCoAPlus = 1;
    }
    //CoSIso
    if (HhType == 'I'){
        evo_hh_CoSIso = 1;
    }
}
```



```
if (HhType_previous == 'I'){
    //IsoAut
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 0))
        || ((HhSize == 4) && (NChildren == 1)) || ((HhSize == 5)
            && (NChildren < 3)) || (HhSize == 6)))){
        evo_hh_IsoAut = 1;
    }
    //IsoCoA
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 1))
        || ((HhSize == 4) && (NChildren == 2)) || ((HhSize == 5)
            && (NChildren == 2))))) {
        evo_hh_IsoCoA = 1;
    }
    //IsoCoS
    if (HhType == 'C'){
        evo_hh_IsoCoS = 1;
    }
    //IsoMon
    if (HhType == 'N'){
        evo_hh_IsoMon = 1;
    }
}

if (HhType_previous == 'N'){
    //MonAlter
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 0))
        || ((HhSize == 4) && (NChildren == 1)) || ((HhSize == 5)
            && (NChildren < 3)) || (HhSize == 6)))){
        if (HhSize > HhSize_previous)
            evo_hh_MonAlterPlus = 1;
        else {
            if (HhSize < HhSize_previous)
                evo_hh_MonAlterMinus = 1;
            else
                evo_hh_MonAlter = 1;
        }
    }
    //MonCoA
```

```
    if ((HhType == 'F') && (((HhSize == 3) && (NChildren == 1))
        || ((HhSize == 4) && (NChildren == 2)) || ((HhSize == 5)
        && (NChildren == 2)))){
        if (HhSize > HhSize_previous)
evo_hh_MonCoAPlus = 1;
        else {
if (HhSize < HhSize_previous)
    evo_hh_MonCoAMinus = 1;
else
    evo_hh_MonCoA = 1;
        }
    }
//MonCoupl
if (HhType == 'C'){
    evo_hh_MonCoupl = 1;
}
//MonIso
if (HhType == 'I'){
    evo_hh_MonIso = 1;
}
//MonMon
if (HhType == 'N'){
    if (HhSize > HhSize_previous)
evo_hh_MonMonPlus = 1;
    else {
if (HhSize < HhSize_previous)
    evo_hh_MonMonMinus = 1;
    }
    }
}
```

C.1.6 Implémentation de la position au sein du ménage

```
//initialisation des variables représentant la position dans le ménage
int hhrel_2 = 0;
int hhrel_3 = 0;
int hhrel_9 = 0;
```

```
//initialisation de la position de l'individu au sein du ménage
if (HhRelationship == 'M')
    hhrel_2 = 1;
else if (HhRelationship == 'A')
    hhrel_3 = 1;
else if (HhRelationship == 'C')
    hhrel_9 = 1;

//coefficients de régression par position
float p_hhrel_2 = -0.0326;
float p_hhrel_3 = -0.8676;
float p_hhrel_9 = -0.3549;
```

C.1.7 Implémentation de l'évolution de la position au sein du ménage

```
//initialisation des variables représentant l'évolution
de la position dans le ménage
int evo_hhrel_12 = 0;
int evo_hhrel_13 = 0;
int evo_hhrel_19 = 0;
int evo_hhrel_21 = 0;
int evo_hhrel_23 = 0;
int evo_hhrel_29 = 0;
int evo_hhrel_31 = 0;
int evo_hhrel_32 = 0;
int evo_hhrel_39 = 0;
int evo_hhrel_91 = 0;
int evo_hhrel_92 = 0;
int evo_hhrel_93 = 0;

//previous_hhrel_1
if (HhRelationship_previous == 'H'){
    if (HhRelationship == 'M')
        evo_hhrel_12 = 1;
    else {
        if (HhRelationship == 'A')
            evo_hhrel_13 = 1;
        else {
```

```
if (HhRelationship == 'C')
    evo_hhrel_19 = 1;
    }
    }
}
else {
    //previous_hhrel_2
    if (HhRelationship_previous == 'M'){
        if (HhRelationship == 'H')
evo_hhrel_21 = 1;
        else {
if (HhRelationship == 'A')
    evo_hhrel_23 = 1;
else {
    if (HhRelationship == 'C')
        evo_hhrel_29 = 1;
}
    }
    }
    else {
        //previous_hhrel_3
        if (HhRelationship_previous == 'A'){
if (HhRelationship == 'H')
    evo_hhrel_31 = 1;
else {
    if (HhRelationship == 'M')
        evo_hhrel_32 = 1;
    else {
        if (HhRelationship == 'C')
            evo_hhrel_39 = 1;
    }
}
    }
    else {
//previous_hhrel_9
if (HhRelationship_previous == 'C'){
    if (HhRelationship == 'H')
        evo_hhrel_91 = 1;
    else {
```

```
    if (HhRelationship == 'M')
        evo_hhrel_92 = 1;
    else {
        if (HhRelationship == 'A')
            evo_hhrel_93 = 1;
        }
    }
}

    }
}

//coefficients de régression pour chaque évolution de la position
float p_evo_hhrel_12 = -0.1419;
float p_evo_hhrel_13 = -0.3373;
float p_evo_hhrel_19 = 0.0449;
float p_evo_hhrel_21 = -0.0449;
float p_evo_hhrel_23 = -0.3872;
float p_evo_hhrel_29 = -0.1396;
float p_evo_hhrel_31 = -0.6197;
float p_evo_hhrel_32 = -0.2858;
float p_evo_hhrel_39 = -0.5364;
float p_evo_hhrel_91 = -0.3801;
float p_evo_hhrel_92 = -0.4479;
float p_evo_hhrel_93 = -0.0605;
```

C.1.8 Implémentation de la migration précédente

```
int hasMoved = 0; //pour l'annee H, on ne sait pas s'il a demenage l'annee H-1

if (this->migration == true) //migration l'annee precedente
    hasMoved = 1;

float p_hasMoved = 0.6269; //si l'individu a demenage
```

C.1.9 Implémentation du niveau d'éducation

```
//initialisation des variables représentant les niveaux d'éducation
int education_sec_inf = 0;
int education_sec_sup = 0;
```

```
int education_prim = 0;
int education_nr = 0;

//initialisation du niveau d'éducation de l'individu
if (this->_education == 'S')
    education_sec_sup = 1;
else if (this->_education == 'P')
    education_prim = 1;
else if (this->_education == '0')
    education_prim = 1;

//coefficients de régression par niveau d'éducation
float p_education_sec_inf = -0.0360;
float p_education_sec_sup = 0.0214;
float p_education_prim = -0.1240;
float p_education_nr = -0.1026; //no reply
```

C.1.10 Implémentation du type d'occupation du logement

```
//initialisation des variables représentant le mode d'occupation du logement
int housingtt_lodger_private = 0;
int housingtt_lodger_social = 0;
int housingtt_free = 0;
int housingtt_nr = 0;

//initialisation du mode d'occupation du logement
if (agentsHh.getAgent(this->getHhId())->getHousingtt() == 'L')
    housingtt_lodger_private = 1;

//coefficients de régression par mode d'occupation du logement
float p_housingtt_lodger_private = 0.6615; //private housing lodger
float p_housingtt_lodger_social = -0.3448; // social housing lodger
float p_housingtt_free = 0.2254; //free housing
float p_housingtt_nr = 0.2369; //no reply
```